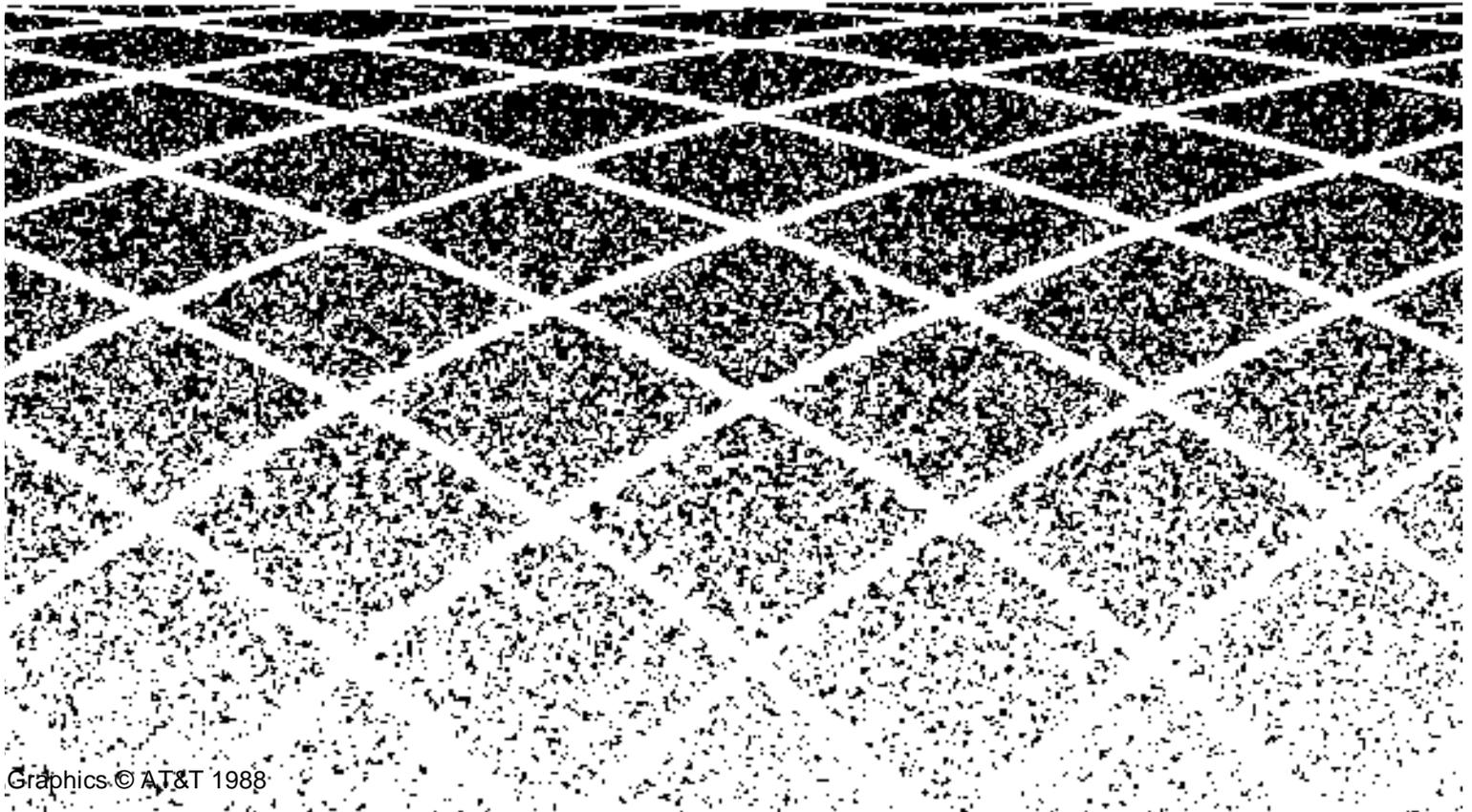




585-310-228  
Issue 1  
March, 1995

**Intuity CONVERSANT  
Voice Information System  
Version 5.0  
Speech Development**





---

# Contents

---

<b>Table of Contents</b>	i
--------------------------	---

---

<b>About This Book</b>	xiii
■ Purpose	xiii
■ Intended Audiences	xiv
■ How This Book Is Organized	xiv
■ How to Use This Book	xv
■ Conventions Used in This Book	xvi
■ Related Resources	xviii
■ Technical Updates	xviii
■ Trademarks and Service Marks	xix
■ How To Make Comments About This Book	xix

---

<b>1</b>	<b>Overview of Speech on the Intuity CONVERSANT VIS</b>	1-1
	■ What's in This Chapter	1-1
	■ Understanding the Speech Filesystem	1-2
	Default Speech Directory	1-2
	Specifying a Speech Directory	1-2
	Adding a Second Speech Disk	1-3
	Recovering from a Corrupted Speech Disk	1-3
	Changing the Size of the Speech Filesystem	1-3
	Creating and Removing Filesystems	1-3
	Displaying Filesystem Attributes	1-4
	■ Underlying Speech Concepts	1-5
	Defining Phrases	1-5
	Defining Phrase Tags	1-5
	Defining Phrase Numbers	1-6
	Defining Talkfiles	1-6
	Defining Speech Files	1-6
	Defining Speech Filesystem	1-6

---

# Contents

■ Overview of Speech Development Features	1-7
Script Builder	1-7
Text-to-Speech	1-8
Graphical Speech Editor	1-9
Speech Recognition	1-10
WholeWord Speech Recognition	1-10
Standard Versus Custom Vocabulary	1-10
Additional Requirements	1-10
FlexWord Speech Recognition	1-11
FlexWord Toolkit	1-11
Additional Requirements	1-11
Form Filler Plus	1-12
■ Summary of Speech Development	1-14

---

<b>2</b>	<b>Developing Speech</b>	2-1
■	What's in This Chapter	2-1
■	Overview of Creating Speech	2-2
■	Determining the Transaction	2-3
■	Planning the Voice Script	2-4
■	Writing the Voice Script	2-6
	Using Framing in Voice Scripts	2-6
	Placing Frame Words	2-7
	Considering Certain Speech Sounds	2-8
	Considering Voiceless Stops and Sounds	2-8
	Considering Speech Inflections	2-8
■	Selecting a Speech Development Method	2-9
	Hiring a Professional Speaker	2-9
	Producing Self-Recorded Custom Speech	2-10
	Purchasing the Standard Speech Package	2-11
	Purchasing an AT&T Custom Speech Package	2-11
	Advantages of AT&T Custom Speech	2-11
	Using Text-to-Speech	2-12
	Using the Graphical Speech Editor	2-13
	Sharing Speech	2-13

---

## Contents

Importing Speech	2-14
■ Recording Speech	2-15
Using a Professional Speaker	2-15
Environmental Conditions	2-15
Equipment and Recording Parameters	2-15
Equipment Specifications	2-15
Recording the Spoken Phrases on Magnetic Tape	2-16
Recording Natural Sounding Speech	2-16
Recording Sets of Related Words	2-16
Recording Alpha and Numeric Characters with Inflections	2-17
Recording Speech with Script Builder	2-18
Purchasing a Custom Speech Package from AT&T	2-18
■ Digitizing Speech Phrases	2-19
■ Installing Speech	2-20
Installing a Script Builder or TSM Application	2-20
Installing Speech in an IRAPI Application	2-20

---

<b>3</b>	<b>Editing Speech</b>	3-1
■	What's in This Chapter	3-1
■	Introduction to the Graphical Speech Editor	3-2
	Making Changes with GSE	3-2
	Supported Speech Formats	3-2
	Working with Segments of Speech	3-2
■	Accessing the Graphical Speech Editor	3-3
	Installing and Administering the Mouse	3-3
	Installing the Graphical Speech Editor	3-3
	Initiating the Graphical Speech Editor	3-4
■	Graphical Speech Editor User Interface	3-5
	Title Bar	3-5
	Menu Bar	3-6
	File Menu	3-6
	Voice Menu	3-7
	Edit Menu	3-8

---

# Contents

Help Menu	3-8
Tool Bar	3-9
Speech Display Area	3-12
Scroll Bar	3-13
Information Bar	3-13
■ Editing a Speech File with GSE	3-15
Using the Mouse	3-15
Opening a File	3-16
Setting the Marker	3-17
Selecting a Segment of Speech	3-18
Listening to Speech	3-19
Cutting and Pasting Speech	3-19
Copying and Pasting Speech	3-21
Adjusting the Volume	3-22
Undoing Changes	3-23
Saving Changes	3-23
Exiting	3-24
■ Converting Speech Files	3-25
Retrieving Files from Speech Pools	3-26
Example	3-26
Restoring Files to Speech Pools	3-27
Example	3-28
Retrieving Files from the Speech Filesystem	3-29
Example	3-30
Restoring Files to the Speech Filesystem	3-31
Example	3-31
■ Using Script Builder for Editing	3-32
■ Using Third-Party Speech Editing Systems	3-33
Retrieving Speech Files from the Speech Filesystem	3-34
Converting Speech File Formats	3-34
Transferring Speech Files	3-35
Using the Audio Works Station	3-37
Retrieving Speech from the VIS	3-38
Editing Speech in Audio Works Station	3-38
Transferring the Speech to the VIS	3-39

---

# Contents

---

<b>4</b>	<b>Recognizing Speech Input</b>	4-1
■	What's in This Chapter	4-1
■	Introduction to WholeWord Speech Recognition	4-2
■	Components of WholeWord Speech Recognition	4-2
	International Support	4-2
	Bilingual WholeWord Speech Recognition Capabilities	4-2
	Standard Vocabulary	4-2
	Standard WholeWord Speech Recognition Types	4-4
	Models	4-6
	Grammars	4-6
	Custom Vocabulary	4-7
	Custom Recognition Types	4-7
	Custom DIPS	4-8
	Word Spotting	4-8
	Speaker Independence	4-9
	Phrase Screening (Rejection)	4-9
	Recognition Confirmation	4-9
	Barge-In	4-9
	Example 1 — Discouraging Barge-In	4-10
	Example 2 — Discouraging Barge-In	4-10
	Example 3 — Encouraging Barge-In	4-10
	Dual Tone Multifrequency (DTMF) Support	4-11
	Connected Digit Recognition	4-11
	Fixed-Length versus Variable-Length Connected Digits	4-12
■	WholeWord Speech Recognition Accuracy	4-13
	Positive Influences on Wholeword Speech Recognition Accuracy	4-13
	Negative Influences on Wholeword Speech Recognition Accuracy	4-16
■	How WholeWord Speech Recognition Works	4-19
	Speech Input from Caller	4-19
	The Recognizer	4-19
	Computational Processing	4-20

---

# Contents

Further Processing	4-20
■ Introduction to FlexWord Speech Recognition	4-22
Defining Phonemes	4-22
■ Components of FlexWord Speech Recognition	4-24
AT&T Custom Vocabulary Service	4-24
FlexWord Toolkit	4-24
Extended Vocabularies	4-24
Keyword Spotting	4-25
Phrase Screening (Rejection)	4-25
■ Limitations of FlexWord Speech Recognition	4-25
Barge-In Not Supported	4-25
Lack of International Support	4-25
■ Preparing a FlexWord Application	4-26
Designing the FlexWord Application	4-26
Using Wordlists in Scripts	4-26
■ Creating Wordlists	4-30
Purchasing an AT&T Custom Speech Package	4-30
Using the FlexWord Toolkit to Create Your Wordlists	4-30
Hardware and Software Requirements	4-30
Configuring Hardware for the FlexWord Toolkit	4-31
Creating and Modifying FlexWord Wordlists	4-34
Adding Words and Phrases to a Vocabulary	4-39
Deleting Words and Wordlists from a Vocabulary	4-40
Copying Wordlists	4-41
■ FlexWord Vocabulary Administration	4-42
Installing Wordlists	4-42
Activating Wordlists	4-43
Deactivating Wordlists	4-43
Removing Wordlists	4-44
■ FlexWord Speech Recognition Accuracy	4-45
Positive Influences on Recognition Accuracy	4-45
Negative Influences on Recognition Accuracy	4-48
■ WholeWord Versus FlexWord Accuracy	4-50

---

# Contents

---

<b>5</b>	<b>Putting It Together</b>	5-1
	■ What's in This Chapter	5-1
	■ Getting the Most Out of WholeWord	5-2
	Combining WholeWord Speech Recognition and Touch Tone	5-2
	■ Getting the Most Out of FlexWord	5-3
	Building a FlexWord Vocabulary	5-3
	Choosing a FlexWord Vocabulary	5-3
	Caller Error	5-4
	Word Length	5-4
	Word Meaning	5-4
	Prompting for Input	5-4
	Sound of a Word	5-4
	Synonyms	5-4
	Examples of FlexWord Applications	5-5
	Menu-Based Applications	5-5
	Applications Requiring Large Wordlists	5-5
	■ Getting the Most out of Text-to-Speech	5-6
	■ Using FlexWord Speech Recognition and Text-to-Speech	5-7
	■ Using Text-to-Speech and Prerecorded Speech	5-8
	■ Using WholeWord and FlexWord Together	5-10
	■ WholeWord versus FlexWord	5-12

---

<b>A</b>	<b>Standard Speech Phrases</b>	A-1
	■ What's in This Appendix	A-1
	■ Standard Speech Phrases	A-2

---

<b>B</b>	<b>Speech File Formats</b>	B-1
	■ What's in This Appendix	B-1
	■ Speech File Formats	B-2

---

# Contents

More About PCM Speech File Formats	B-2
Linear PCM Format	B-2
Mu-law PCM Format	B-2
More About ADPCM Speech File Formats	B-2
■ More About Speech Files	B-4

---

<b>C</b>	<b>Computing Channel Numbers</b>	C-1
■	What's in This Appendix	C-1
■	Computing Channel Numbers	C-2

---

<b>D</b>	<b>Advanced Text-to-Speech Features</b>	D-1
■	What's in This Appendix	D-1
■	Introduction	D-2
■	Using Escape Sequences	D-2
	Silence Delays	D-2
	Speaking Rate	D-4
	Additional Text Classes	D-5
	Changing Class Detection	D-6
	Setting Specific Class Detector Modes	D-6
	Sample Application Using Escape Sequences	D-8
	Changing the Volume Level	D-14
	Relating the Incoming Text Volume to Outgoing Text Volume	D-16

---

## **Contents**

---

<b>ABB</b>	<b>Abbreviations</b>	ABB-1
------------	----------------------	-------

---

<b>GL</b>	<b>Glossary</b>	GL-1
-----------	-----------------	------

---

<b>IN</b>	<b>Index</b>	IN-1
-----------	--------------	------

---

# Contents

---

## About This Book

---

### Purpose

This book, *Intuity CONVERSANT Voice Information System Version 5.0 Speech Development*, 585-310-228, is a reference on speech development for applications developed on the AT&T INTUITY™ CONVERSANT® System. An *application* is made up of several components, one of which is speech, that provide an automated version of the communication between a caller and an agent. Another component of an application, which is mentioned throughout this book, is the *transaction*. The transaction is comprised of the exchanges between the caller and the VIS. The *script* is the actual set of instructions for the VIS to follow during a transaction. These terms are used throughout this book.

Most applications, whether generated with Script Builder or developed using the transaction state machine (TSM) script-level language and/or C language, involve some form of speech (for example, playing recorded phrases, recognizing speech input, recording spoken phrases, or playing synthesized speech). An application may use one, some, or all forms of speech the Intuity CONVERSANT VIS has to offer. This book offers information about everything that is involved in speech development — from determining and planning your transaction to installing the speech. The main focus of this book, though, is on detailed information about the features available for developing speech.

For detailed information about creating an application with Script Builder, see *Intuity CONVERSANT Voice Information System Version 5.0 Script Builder*, 585-310-727. For detailed information about creating an application with the TSM script-level language, see *Intuity CONVERSANT Voice Information System Version 5.0 Application Development*, 585-310-227. For detailed information about creating an application with the Intuity Response Application Programming Interface (IRAPI), see *Intuity CONVERSANT VIS V5.0 IRAPI Programming Guide*, 585-310-226.

## **Intended Audiences**

---

The primary audiences for the *Intuity CONVERSANT VIS V5.0 Speech Development* book are as follows:

- End customer developers  
This group is responsible for creating and maintaining applications in the Intuity CONVERSANT VIS environment.
- Custom application developers  
This group is responsible for creating applications to be used in the VIS environment for end-user customers. This audience includes any AT&T custom application developers.
- Application distributors  
This group distributes and implements applications for end-users. This audience includes Independent Software Vendors (ISV) and Voice Processing CoMarketers (VPC).

## **How This Book Is Organized**

---

This book is organized into the following chapters:

- Chapter 1, "Overview of Speech on the Intuity CONVERSANT VIS"  
This chapter provides an overview of speech on the Intuity CONVERSANT VIS. Information about the speech file system and the different parts of speech are discussed. An overview of all of the speech-related features available for the development of applications on the VIS is also covered in this chapter.
- Chapter 2, "Developing Speech"  
This chapter defines and describes the procedures involved in producing speech — from determining the transaction and planning the script to recording and encoding the speech.
- Chapter 3, "Editing Speech"  
This chapter includes information on using the Graphical Speech Editor, Script Builder, and third-party speech editing systems for editing speech.
- Chapter 4, "Recognizing Speech Input"  
This chapter includes an introduction to speech recognition, along with a detailed discussion of both WholeWord Speech Recognition and FlexWord Speech Recognition. Also included is a section on creating vocabularies and improving the accuracy of speech recognition.
- Chapter 5, "Putting It Together"  
This chapter discusses the speech-related features and how to make them work work well together in your application.

- Appendix A, "Standard Speech Phrases"  
This appendix provides a complete listing of all standard speech phrases.
- Appendix B, "Speech File Formats"  
This appendix includes information about speech files and speech file formats, including Pulse Code Modulation (PCM) and Adaptive Differential Pulse Code Modulation (ADPCM).
- Appendix C, "Computing Channel Numbers"  
This appendix includes the procedure for computing channel numbers for the GSE and the FlexWord Toolkit.
- Appendix D, "Advanced Text-to-Speech Features"  
This appendix discusses several ways to customize synthesized speech, including using escape sequences to add silence delays, change the speaking rate, and mark text as belonging to a specific text category.

This book also includes a list of Abbreviations, a Glossary, and a cross-referenced Index.

## **How to Use This Book**

---

This book covers two overall concepts regarding speech on the VIS:

- Developing speech that prompts callers in your application
- Recognizing speech input or caller input during a transaction

This book is organized in a logical, procedural manner in which you learn about speech on the Intuity CONVERSANT VIS (Chapter 1, "Overview of Speech on the Intuity CONVERSANT VIS"), followed by how to create speech (Chapter 2, "Developing Speech") and edit speech (Chapter 3, "Editing Speech"). Next, the focus turns to recognizing caller input and using speech recognition features (Chapter 4, "Recognizing Speech Input"). Finally, you learn how to effectively use speech-related features together (Chapter 5, "Putting It Together").

## Conventions Used in This Book

---

The following typographic conventions are used in this book:

- Terminal keys

- Terminal keys are shown in rounded boxes. For example, an instruction to press the enter key is shown as

Press `ENTER`.

- Function keys (also known as *soft* keys) are shown in rounded boxes followed by the function of that key in parentheses. For example, an instruction to press function key F3 is shown as

Press `F3` (CHOICES).

- Two or three keys that you press at the same time (that is, you hold down the first key while pressing the second and/or third key) are shown as a series of rounded boxes. For example, an instruction to press and hold `ALT` while typing the letter `d` is shown as

Press `ALT` `D`.

- User input

- The word *enter* means to type a value and press `ENTER`. For example, an instruction to type `y` and press `ENTER` is shown as

Enter `y` to continue.

- The word *type* means to press the key or sequence of keys specified. For example, an instruction to type `y` is shown as

Type `y` to continue.

Do *not* press `ENTER` after you type the value specified.

- The word *select* means to move to the desired menu item using the arrow keys and press `ENTER`. For example, an instruction to select an item from a menu and press `ENTER` is shown as

Select Configuration Management from the Voice System Administration menu.

- Information that you enter or type from your terminal keyboard is shown in **bold** type; for example

Enter **root** at the `Console Login` prompt.

- Command and file names and their parameters are shown in **bold** type. Variable parameters are shown in ***bold italic*** type when they are part of a user input and in *regular italic* type when they are not. All are illustrated in the following example:

Use the **print** command to print your report. The command syntax is **print *reportname***, where *reportname* is the name of the report to be printed.

- Screen displays

- Information that is displayed on your terminal screen — including screen displays, prompts, script code, and system messages — is shown in `typewriter-style` type; for example

```
Installation is in progress -- do not remove
the floppy disk.
```

- The sequence of menu options that you must select to display a specific screen is shown as follows:

Begin at the CONVERSANT Administration menu, and select the following sequence:

```
> Voice System Administration
```

```
> Equipment
```

In this example, you would first access the CONVERSANT Administration menu. Then you would select the Voice System Administration option to display the Voice System Administration menu. From that menu, you would select the Equipment option to display the Voice Equipment screen.

- The screens shown in the Intuity CONVERSANT library are only examples. Your screens may not appear exactly as illustrated.

## **Related Resources**

---

The following books should be used in conjunction with this book:

- *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727

This book describes the VIS Script Builder and how to use it to design applications to be used in the VIS environment. This book will be referred to for incorporating speech into your Script Builder application.

- *Intuity CONVERSANT VIS V5.0 Application Development*, 585-310-227

This book provides information about designing software applications and writing programs that integrate the application software and generic software. This book will be referred to for incorporating speech into your TSM application.

- *Intuity CONVERSANT VIS V5.0 IRAPI Programming Guide*, 585-310-226

This book provides a description of the application development process using the Intuity Response Application Programming Interface (IRAPI). This book will be referred to for incorporating speech into your IRAPI application.

Refer to *Intuity CONVERSANT VIS Documentation Guide*, 585-310-020, for a complete list of associated documentation.

## **Technical Updates**

---

Every effort was made to ensure that the information contained in these books is technically accurate, and will guide readers in the normal operation of the system. There are instances however, when the Intuity CONVERSANT VIS Version 5.0 product behaves differently than is documented in the core library.

To help with this, an online bulletin board is available to all Intuity CONVERSANT VIS Version 5.0 customers that provides supplemental information about this product in an electronic, E-mail format. These updates include hints, tips, and exception conditions about all aspects of the Intuity CONVERSANT VIS Version 5.0 product that were discovered after the core library was published.

This service is called Access, and is available 24 hours-a-day, seven days-a-week to anyone who subscribes to it. To begin receiving electronic Intuity CONVERSANT VIS Version 5.0 Access articles, call 1-800-242-6005, and ask for department 186.

## **Trademarks and Service Marks**

---

The following trademarked products are mentioned in the Intuity CONVERSANT VIS Version 5.0 documentation:

- AUDIX, CONVERSANT, DEFINITY, 5ESS, and 4ESS are registered trademarks of AT&T.
- Voice Power, Intuity, and FlexWord are trademarks of AT&T.
- UnixWare is a registered trademark of Novell, Inc.
- ORACLE, ORACLE\*Terminal, OBJECT\*SQL, SQL\*FORMS, SQL\*Menu, SQL\*Net, SQL\*Plus, PRO\*C, and SQL\*Report Writer are trademarks of the Oracle Corporation.
- IBM is a registered trademark of International Business Machines.
- CLEO and LINKix are trademarks of CLEO Communications.
- Hayes and Smartmodem are trademarks of Hayes Microcomputer Products, Inc.

## **How To Make Comments About This Book**

---

A reader comment card is included following the title page of this book. While we have tried to make this book fit your needs, we are interested in your suggestions for improving it and urge you to complete and return a reader comment card.

If the reader comment card has been removed from this book, please send your comments to:

AT&T  
Product Documentation Development  
Room 22-2C11  
11900 North Pecos Street  
Denver, Colorado 80234

Please include the name and document number of this book.



---

# Overview of Speech on the Intuity CONVERSANT VIS

# 1

---

## What's in This Chapter

This chapter provides an overview of speech on the Intuity CONVERSANT Voice Information System (VIS). It provides information about the speech filesystem, speech terminology, and an overview of the speech-related features available for speech development.

## Understanding the Speech Filesystem

All speech to be played as part of your application resides as UNIX files in a mounted UNIX filesystem. Speech filesystems reside in **/home2**. With the Intuity CONVERSANT VIS, you are able to define where in **/home2** you want your speech files stored.

Each speech phrase requires a minimum of 16 Kbytes of space. Depending on your coding rate, a 16-Kbyte block holds different amounts of speech. Refer to Table 1-1 to estimate how much speech is in each speech phrase block.

**Table 1-1. Coding Rates and Speech Phrase Blocks**

---

<b>Coding Rate</b>	<b>Seconds of Speech Per 16 Kbyte Block</b>
SBC16	8
SBC24	6
ADPCM32	4
ADPCM64	2

---

Refer to Appendix B, "Speech File Formats", for detailed information about coding rates. Refer to Chapter 4, "Features," in *Intuity CONVERSANT VIS V5.0 System Description*, 585-310-225, in the sections titled "Speech Storage Capacities" and "Speech Channel Capacities" for more information about the encoding methods and capacities.

## Default Speech Directory

The default speech directory is designated as **/home2/vfs/talkfiles**, which is a UNIX directory for storing speech. It is organized into 8-Kbyte blocks, which allows for quick and efficient retrieval of speech files.

## Specifying a Speech Directory

To change the speech directory in which your speech files are stored (from the default), perform the following procedure:

1. Stop the voice system.

Refer to Chapter 3, "Configuration Management," of *Intuity CONVERSANT VIS V5.0 Operations*, 585-310-550, in the section titled "System Control" for instructions on how to stop the voice system.

2. Access the **/vs/data/irAPI.rc** file.

3. Add the following entry, where *<directory>* is the full path of the new directory where you want to store your speech files:

**SPEECHDIR=<directory>**

The SPEECHDIR variable specifies the new directory.

4. Restart the voice system.

Refer to Chapter 3, "Configuration Management" of *Intuity CONVERSANT VIS V5.0 Operations*, 585-310-550, in the section titled "System Control" for instructions on how to start the voice system.

 **NOTE:**

The speech administration tools (for example, **list**, **add**, **copy**, **erase**, and **vdf**) are only available for use with speech files stored in the speech directory defined by the SPEECHDIR variable in the **irAPI.rc** file.

### Adding a Second Speech Disk

If your system has more than 72 channels (telephone network connections), you must add a second disk for storing speech. If your system has less than 72 channels, you have the option of adding a second disk. Refer to Appendix C, "Disk Operations," in *Intuity CONVERSANT VIS V5.0 Maintenance*, 585-310-153, for information on how to add a second disk. The second disk in your system is used to store speech filesystems that reside in **/home3**.

### Recovering from a Corrupted Speech Disk

For information on recovering from a corrupted speech disk, refer to Chapter 4, "Common Maintenance Procedures," in *Intuity CONVERSANT VIS V5.0 Maintenance*, 585-310-153.

### Changing the Size of the Speech Filesystem

With the Veritas Advanced Filesystem, you can change the size of your speech filesystem without having to reload UNIX. Refer to Chapter 5, "OA&M Menu Interface Operation" in the *System Administrator's Guide*, 585-350-906, in the section titled "Basic Filesystem Operations" for detailed information about increasing and decreasing the size of your filesystem.

### Creating and Removing Filesystems

With the Veritas Advanced Filesystem, you can create and remove speech filesystems in Intuity CONVERSANT VIS Version 5.0. Refer to Chapter 5, "OA&M Menu Interface Operation" in the *System Administrator's Guide*, 585-350-906, in the section titled "Basic Filesystem Operations" for detailed information about creating and removing filesystems.

## **Displaying Filesystem Attributes**

---

With the Veritas Advanced Filesystem, you can display the contents of your filesystems. Refer to Chapter 5, "OA&M Menu Interface Operation" in the *System Administrator's Guide*, 585-350-906, in the section titled "Basic Filesystem Operations" for detailed information about displaying the contents of your filesystems.

## **Underlying Speech Concepts**

---

Before you begin developing speech for your application, you should understand the underlying concepts behind speech on the VIS.

### **Defining Phrases**

---

A phrase is a unit of speech (for example, letter, number, word, sentence, or a paragraph) that is spoken to the caller. Examples of phrases include a welcome message, a bank balance, or the name of a month. Every phrase in an application is identified by a phrase tag or phrase number. A script speaks a phrase to callers by referencing either the phrase tag or the phrase number in the application. Refer to the following sections "Defining Phrase Tags" and "Defining Phrase Numbers" for more information.

### **Defining Phrase Tags**

---

A phrase tag is a string of up to 50 characters that identifies the contents of a phrase used by an application script. In other words, a phrase tag identifies a specific phrase. When you define a message to be played during a transaction, you specify a given phrase by its phrase tag (as opposed to its content). The two types of phrase tags are as follows:

— Standard speech

- Dollars
- Cents
- Time
- Weekdays
- Months
- Numbers

The Standard Speech package includes prerecorded speech phrases corresponding to the above types of phrases. For a list of the speech phrases that come with the Standard Speech package, refer to Appendix A, "Standard Speech Phrases".

— Custom speech

Custom phrase tags are designed specifically for the application you are developing and are usually more than one word in length. For example,

- "Your account balance is"
- "Please enter your 5-digit account number"
- "The current interest rate is"

Script Builder uses predefined phrase tags for spoken output, such as digits and letters in various inflections. For more information about the Script Builder predefined phrase tags, refer to Chapter 10, "Speech Administration," in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727.

 **NOTE:**

Predefined phrase tags begin with a colon (:). Therefore, do not use a colon as the first character in any phrase tag.

## Defining Phrase Numbers

A phrase number is a number that identifies the contents of a phrase used by an application script. A script speaks a phrase to callers by referencing either the phrase tag or the phrase number. A phrase number is assigned to your phrase when you add the phrase to your script (for example, adding a phrase through a **Prompt & Collect** action in your Script Builder application).

## Defining Talkfiles

A talkfile is a list of phrases usually associated with an application script. Talkfiles are stored under the directory **/speech/talk**. All of these files have a **.pl** extension. The first line in the file shows the talkfile number. The rest of the file displays the phrases (as they were entered in the application) preceded by their phrase numbers. The actual phrases are located in the speech filesystem.

Each talkfile may contain up to 65535 phrases. Your Intuity CONVERSANT VIS system can have up to 16384 talkfiles, although your system is limited by the size of your hard disk. The talkfile number and phrase tag or phrase number together uniquely identify a phrase.

## Defining Speech Files

A speech file is a file containing an encoded speech phrase. Speech files can be stored anywhere, although the default speech filesystem is located in the **/home2/vfs/talkfiles** directory.

## Defining Speech Filesystem

A speech filesystem is a mounted UNIX filesystem where speech resides and is defined in the **irAPI.rc** file. Only one speech filesystem can be active at a given time. Refer to the section, "Understanding the Speech Filesystem" at the beginning of this chapter for more information about Intuity CONVERSANT VIS filesystems.

## **Overview of Speech Development Features**

---

Several speech development features are available for the creation, editing, recognition, and inclusion of speech in your application. The following sections summarize all of the features available for speech development on the VIS. The available features are as follows:

- "Script Builder"
- "Text-to-Speech"
- "Graphical Speech Editor"
- "WholeWord Speech Recognition"
- "FlexWord Speech Recognition"
- "Form Filler Plus"

### **Script Builder**

---

Nearly all applications involve playing recorded speech to the caller. Script Builder provides one way of producing this recorded speech. Script Builder allows you to design and develop VIS applications that automate most functions performed by operators or agents. For additional general information about producing speech with Script Builder, see Chapter 2, "Developing Speech".

## **Text-to-Speech**

---

Text-to-Speech (TTS) is another way of producing speech for your application. TTS takes text as input and produces synthesized speech spoken in a male voice. You can use TTS to produce text for prompts, text retrieved from a database or host, or text file spoken in an application. It is easy to use TTS in your application — you simply type the text to be spoken into your Script Builder application or your TSM script via the keyboard. Refer to Chapter 6, “Using Optional Features with Script Builder,” in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for more information about using TTS with Script Builder. Refer to Chapter 3, “Script Instructions,” in *Intuity CONVERSANT VIS V5.0 Application Development*, 585-310-227, for more information about using TTS in your TSM application.

TTS is usually used to speak text that changes frequently, for example:

- Large databases of names and addresses
- Dynamic text, such as electronic mail
- Prototype applications, where it eliminates the need to record custom phrases

The requirements for Text-to-Speech are as follows:

- Text-to-Speech software package
- AYC9 Signal Processor (SP) circuit card

For the software installation procedure, see *Intuity CONVERSANT VIS V5.0 Software Installation*, 585-310-151. For instructions on how to install the AYC9 SP, see the Intuity CONVERSANT VIS Version 5.0 Hardware book specific for your platform.

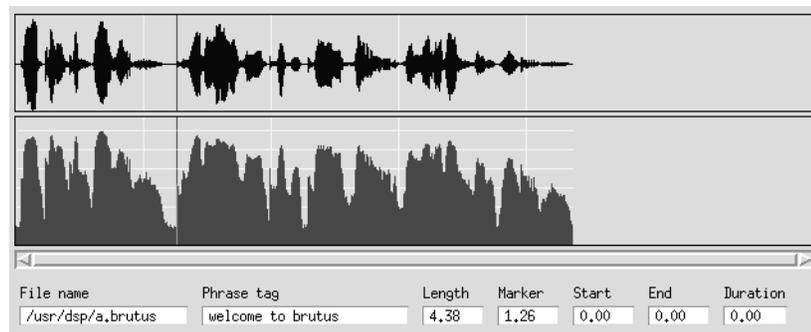
---

## Graphical Speech Editor

---

The Graphical Speech Editor (GSE) provides a graphical interface for creating and editing speech files. This user-friendly interface includes menus that you can access to perform different functions associated with retrieving, creating, editing, and saving speech files. Figure 1-1 is an example of a speech file in the Graphical Speech Editor. Two speech graphs, envelope and energy, represent the digitized audio signal. This enhances the editing process because it provides the user with a visual representation of the speech file being edited. For more information about the GSE, refer to Chapter 3, "Editing Speech."

---



---

**Figure 1-1. Graphical Speech Editor Layout**

The requirements for the GSE are as follows:

- Graphical Speech Editor software package
- Mouse
- 50mHz CPU

For more information about system capacities and requirements, see *Intuity CONVERSANT VIS V5.0 System Description*, 585-310-225. For information about recording and editing speech with the GSE, refer to Chapter 3, "Editing Speech".

## **Speech Recognition**

---

Speech recognition is an Intuity CONVERSANT VIS capability that allows the VIS to recognize and respond to spoken voice responses from the caller. Speech recognition is offered as either WholeWord Speech Recognition or FlexWord Speech Recognition.

### **WholeWord Speech Recognition**

WholeWord Speech Recognition recognizes entire words, not phonemes or parts of words. The smallest unit of speech that is recognized is a whole word; however, WholeWord Speech Recognition is not restricted to recognizing only one word (for example, "calling card" can be recognized as a whole word).

### **Standard Versus Custom Vocabulary**

AT&T provides a standard WholeWord vocabulary that includes the digits 0-9, "yes," "no," and "oh." The Standard WholeWord vocabulary also provides connected digit recognition. The standard vocabulary is best suited for applications that require callers to respond with "yes" or "no," for example, a survey application, or for number intensive applications such as a banking application that requires callers to enter checking account numbers. You may choose to purchase a custom vocabulary if the standard WholeWord vocabulary does not meet your needs.

### **Additional Requirements**

The requirements for the WholeWord Speech Recognition package are as follows:

- WholeWord Speech Recognition package
- Signal Processing (SP) circuit card
- Companion (CMP) circuit card

For more information on hardware requirements, see *Intuity CONVERSANT VIS V5.0 System Description*, 585-310-225.

For more information on WholeWord Speech Recognition, refer to Chapter 4, "Recognizing Speech Input".

## FlexWord Speech Recognition

FlexWord Speech Recognition recognizes phonemes or parts of words for analyzing and recognizing American English vocabularies.

### NOTE:

FlexWord Speech Recognition only supports the American English language.

Because FlexWord provides a cost-effective way of designing large, customized vocabularies and menu options, it is ideal for word- or phrase-intensive applications. For example, a name dialer, an application which allows employees to speak the name of another employee instead of dialing a telephone extension, could be designed using FlexWord. Since names can be built from phonemes, whole-word data collection processes are not necessary.

## FlexWord Toolkit

The optional FlexWord Toolkit allows you to:

- Create FlexWord wordlists and vocabularies
- Add and delete words and wordlists to an existing vocabulary
- Change the phonetic structure of words in an existing vocabulary
- Hear audible playback of the phonemic representation of a word

Refer to Chapter 4, "Recognizing Speech Input", for detailed information on how to use the FlexWord Toolkit to create words, wordlists, and vocabularies.

## Additional Requirements

The requirements for the FlexWord Speech Recognition package are as follows:

- FlexWord Speech Recognition package
- Signal Processing (SP) circuit card
- Companion (CMP) circuit card

For more information on hardware requirements, see *Intuity CONVERSANT Voice Information System Version 5.0 System Description*, 585-310-225.

For more information about FlexWord Speech Recognition, refer to Chapter 4, "Recognizing Speech Input".

For the installation procedure, see *Intuity CONVERSANT Voice Information System Version 5.0 Software Installation*, 585-310-151.

### NOTE:

You cannot use the FlexWord Toolkit with releases prior to Intuity CONVERSANT VIS Version 5.0. Also, you cannot build wordlists with the

FlexWord Toolkit and move them back to a CONVERSANT VIS Version 4.0 system.

## **Form Filler Plus**

---

The Form Filler Plus feature provides the capability for application scripts to record callers' responses to prompts for later transcription and review. As many as 10 responses may be recorded per call session. Caller responses are recorded and then stored in the Form Filler Plus database, where they may be retrieved at a later time using the Form Retriever transcription script.

Application voice forms which prompt for and record caller input for Form Filler Plus are available through a high-level Script Builder application template provided with the Form Filler Plus package. By simply copying and modifying this template to suit your needs, you may develop a customized Form Filler Plus application. Alternatively, you may use the **FF\_Code** and **FF\_Store** action steps provided with the Form Filler Plus package to develop a customized Form Filler Plus application.

Whether you use the Form Filler application template or the action steps provided with the Form Filler Plus package to develop your own application, the Form Filler Plus package facilitates the development of voice capture and transcription scripts. Form Filler Plus application developers are able to select the coding rate that best suits their application and are given access to the advanced error-handling capabilities available with this feature, including the ability to store partial messages in the event of error or caller hangup.

The Form Retriever transcription script provides an easy-to-use one-key operation. Authorized Form Filler Plus transcribers and reviewers may retrieve caller responses by simply calling the Form Retriever transcription script. Alternatively, authorized Form Filler Plus users may access the Form Retriever by using the **Execute** action access code. The **Execute** action starts a new script on a channel, replacing the script that performed the **Execute** action. The Form Retriever will prompt the user for a mode, password, and Script ID. (Transcribe and review mode passwords may be set when the Form Filler Plus package is installed, whereas the Script ID is determined by the script developer). Caller responses are retrieved by Script ID. The transcriber or reviewer may either choose to hear calls for a particular script by entering a Script ID or may choose to hear calls for the script with the oldest call record by entering an asterisk (\*) to retrieve the script with the oldest record. Once a script is selected, call records are played from oldest to newest and each response recorded in a call record is played from first to last.

Transcribers (using transcribe mode) may play and replay the recorded phrases in a new call record and either delete the record or mark it for review. Once the record is deleted or marked, the next oldest new call record for the Script ID is played.

Reviewers (using review mode) may play and replay a call record that has previously been marked for review by a transcriber. Reviewers may then delete the record or skip it for later review. Once the record is deleted or skipped, the next oldest marked call record for the Script ID is played.

Transcribers or reviewers may use an “undo” command to undo the previous delete, mark, or skip command and return to the previous call record. Although there is no limit to the number of transcribers that may use the application at any one time, only one individual may work on a particular call record at a time. The Form Retriever will bypass records that are being played by other users. Transcribers or reviewers may adjust the pause interval between each phrase that is played from a call record with “slow down” and “speed up” commands.

The number of records deleted or marked for review will be announced at the termination of the transcription or review session, enabling Form Filler Plus users to note their activity.

For detailed information about how to use Form Filler Plus feature, refer to Chapter 6, “Using Optional Features with Script Builder,” in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727.

## **Summary of Speech Development**

Speech development consists of several steps, depending on the features you use. You begin by planning your application. This involves determining how your application can best use the available speech development features. You then record and store the speech, followed by any editing that may be necessary. The other aspect of speech development could involve recognizing caller input received during a transaction. Refer to the following chapters in this book for more information about each of these steps.

---

## Developing Speech

# 2

---

### **What's in This Chapter**

---

This chapter defines and describes the procedures for creating speech — from determining the transaction and planning the script to recording and encoding the speech.

## **Overview of Creating Speech**

---

Speech processing begins with the creation of encoded and digitized speech files for disk storage. The content of each speech file is a single speech phrase that is spoken at some point in an application dialog. A speech phrase can consist of a full sentence, a single word, a specified period of silence, music, or even a tone signal (for example, a "beep") that is specific to an application. You determine the speech phrase content based on the application requirements.

During a call, the individual speech phrases specified in the script are downloaded by the VIS from a hard disk drive to a Tip/Ring (T/R) or Signal Processing (SP) circuit card. The T/R or SP circuit card actually plays the speech.

Processing the recorded speech so that it can be accessed by the VIS involves the following activities:

1. "Determining the Transaction"
2. "Planning the Voice Script" and "Writing the Voice Script"
3. "Selecting a Speech Development Method"
4. "Recording Speech"
5. "Digitizing Speech Phrases"
6. "Installing Speech"

Each of these activities are discussed in the sections that follow.

## **Determining the Transaction**

The application provides the automated version of the communication between the caller and the agent. The transaction is one component of the application that involves the actual exchanges between the caller and the agent.

Before you can begin speech development, you must determine the transaction for the application and at least know the outline of the application that you want to follow, as well as a general idea of what speech phrases/prompts are necessary. For example, you need to decide what type of service you are going to provide, as well as the language and the gender in which the speech will be recorded. Refer to the following for more information:

- *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for information on developing your application using Script Builder.
- *Intuity CONVERSANT VIS V5.0 Application Development*, 585-310-227, for more information on developing your application using the TSM script level language and/or C language.
- *Intuity CONVERSANT VIS V5.0 IRAPI Programming Guide*, 585-310-226, for more information on developing your application using IRAPI.
- *Intuity CONVERSANT VIS V5.0 Application Design Handbook*, 585-310-227, for information about planning your voice response application.

## **Planning the Voice Script**

---

After determining the transaction, you can begin planning the voice script. The voice script includes the exact phrases to be recorded, based on the transaction you have determined. The following are suggestions to consider before writing the voice script:

- Write out every word that you expect to be spoken. Edit the voice script to change any poorly written or repetitive phrases. You want the voice script to read as clearly as possible because professional speakers use the voice script to record the phrases.
- Ensure that changes are written into the voice script if changes are made during recording.
- Track the contents of the voice script by using phrase numbers. Number each phrase in the written voice script.
- Make all commands short and easy to understand. Users tend to remember only the ends of phrases, so place the needed caller action at the end of a phrase. For example, “For account information, press one.”
- Review your voice script to see if the prompts and responses make sense.
- Make prompts clear, but courteous. Remember to welcome users to your company and the system. Thank them at the end.
- Use vocabulary that is understandable and not beyond the scope of your users. For example, do not use computer or programming terminology unless it is familiar to all your users.
- Use the following types of phrases in your voice script:
  - Long phrases that stand alone, for example,  
“Welcome to the Intuity CONVERSANT VIS order entry system.”  
  
Long phrases are easier to speak for a recording because they stand alone.
  - Short phrases that you plan to concatenate, for example,  
“Your balance is “  
“Press 1”

Typically short phrases include phrases that will be used over and over again.

- Anticipate the environment in which the phrases will be used — that is, whether the phrase will be used at the beginning of a sentence, in the middle of the sentence, or at the end of a sentence. The following example shows each use of the word “enter.”

“*Enter* the pound key.” (phrase at the beginning of the sentence)

“You need to *enter* the pound key.” (phrase in the middle of the sentence)

“Please press *enter*.” (phrase at the end of the sentence)

You may plan to use the word “enter” as one phrase, but, you need three recordings of this phrase (one phrase with rising inflection, one with medial inflection, and one with falling inflection). Recording words with emphasis is discussed in the next section, “Writing the Voice Script”.

- Avoid a long string of adjectives. The following is a good description of a poorly designed instruction: “Check the 30-pin, flat ribbon, backplane, bus cable.”

## **Writing the Voice Script**

---

In writing the voice script for a professional speaker, prepare a document that produces the best recordings possible. Mark the target phrases in a way that is easy for the speaker to recognize. Placing quotation marks around the important phrases is helpful. This is called *framing*.

## **Using Framing in Voice Scripts**

---

Human speech is a continuous, uninterrupted signal. It should not be assumed that you can remove a word from one phrase and place that same word in another phrase that is being recorded for a different use. Individual words that you plan to concatenate must be carefully recorded with the proper inflections and sounds framing them.

To achieve a better recording of short words and phrases, use quotation marks to *frame* those words you want to emphasize. For example, to achieve accurate recordings of the word “enter,” use quotation marks in your voice script as follows so that the speaker concentrates on the word “enter:”

“Enter” the pound sign.

Please press “enter.”

The following is an example of a well-prepared voice script that uses framing. The information in quotation marks is the information that the professional speaker should focus on, while the remaining information is the framework.

“Welcome to our telephone information service.”

“To learn more about our investment opportunities, press the star key.”

“This amount represents” the total balance.

“Please enter” two oh one.

You have “a balance of” two hundred dollars.

You can deposit “up to” five hundred dollars.

### Placing Frame Words

Place words and/or phrases before and after the word and/or phrase that you need recorded if possible. These phrases should be familiar phrases that guide the speaker into speaking the word and/or phrase with a certain inflection. For example, if you want an accurate recording of the word “and” with medial inflection, you could, for example, record the word “and” in both of the following frames:

Installing “and” verifying

Cutting “and” pasting

You can remove the words that frame “and” later since they are not needed. These frame words are important, though, because the frame words enable a speaker to speak the word “and” in the necessary context that ensures that it is concatenated properly when used in a phrase.



**NOTE:**

The word “and” is part of the AT&T Standard Speech Package.

### Considering Certain Speech Sounds

Words that end with the < r > or < l > sounds do not make good framing words because those sounds carry over to the next word. In this example,

December "eighth"

"December" is not a good frame word because it ends in an < r > sound, which affects the vowel quality of "eighth."

A better frame word is as follows:

August "eighth"

### Considering Voiceless Stops and Sounds

Voiceless stops are sounds like < p >, < t >, and < k >. Voiceless stops or sounds preceding and following your target word also help you to make an accurate recording. In the example above, the final < t > of "August" provides a silence that makes it easy to isolate "eight." Other voiceless sounds that might work to end or initiate a frame or space are < f > or < s >.

### Considering Speech Inflections

---

Three types of inflection exist with speech phrases:

- Rising inflection

Rising inflection is usually used in questions and at the beginning of some words. For example, when you ask "How can I help you?," the word "you" is spoken with rising inflection. Rising inflection is denoted in Table A-1 of Appendix A, "Standard Speech Phrases" with a question mark (?) that follows the phrase tag.

- Medial inflection

Medial inflection is usually used in the middle of a word or statement. For example, when you speak the number "101," the "0" is spoken with medial inflection. Medial inflection is denoted in Table A-1 of Appendix A, "Standard Speech Phrases" without anything following the phrase tags.

- Falling inflection

Falling inflection is usually used at the end of a word or statement. For example, when you speak "2.0," the "0" is spoken with falling inflection. Falling inflection is denoted in Table A-1 of Appendix A, "Standard Speech Phrases" with a period (.) following the phrase tag.

---

## Selecting a Speech Development Method

---

As an application developer, you have several options from which to choose for including speech in your application. These options are:

- Hiring a professional speaker
- Using the Standard Speech Package\*
- Purchasing a custom speech package from AT&T
- Producing self-recorded custom speech with Script Builder
- Using Text-to-Speech (TTS) \*
- Using the Graphical Speech Editor (GSE)
- Sharing speech already recorded in another application \*
- Importing speech from another application \*

The sections that follow discuss all of the above options.

### Hiring a Professional Speaker

Hiring a professional speaker, such as an actor or radio or television announcer, gives you recorded speech of a high quality. An additional advantage of using a professional speaker is that you may be able to obtain more control and faster response when adding new speech phrases. Consider the following guidelines when choosing a professional speaker:

- Have all phrases prepared for the speaker to read in advance of the recording session. Refer to "Planning the Voice Script" and "Writing the Voice Script" for guidelines on preparing and writing the voice script for the speaker.
- Record and digitize various candidate voices before hiring a speaker to ensure that the encoded quality is good. You may want to listen to several male and female voices to compare the digitized quality.
- Use the same speaker for all speech associated with a specific application.

---

\* These methods do not require you to record speech.

- Make sure that the speaker is able to maintain the following:
  - A constant speaking rhythm and general intonation throughout the recording session  
This ensures that phrases spoken early in the session result in normal-sounding speech when they are concatenated with phrases spoken later in the session
  - A constant acceptable level of volume
  - Clear pronunciation
  - A constant orientation and distance from the microphone
- Ensure that alpha and numeric characters that are to be recorded with rising, medial, and falling inflections are spoken with the appropriate inflections.

If you hire a professional speaker, you can use the GSE to modify the speech phrases for the application script. Refer to Chapter 3, "Editing Speech", for more information about editing speech phrases with the GSE.

Refer to the section, "Recording Speech" later in this chapter for information on setting up a recording session with a professional speaker.

### **Producing Self-Recorded Custom Speech**

You can produce speech with the optional Script Builder package. Script Builder allows you to record and store speech on the hard disk drive. To develop your application, you may want to begin with the standard speech phrases mentioned previously. Again, the standard set includes letters and digits in different speaking inflections and many commonly used phrases, such as the words used to speak dates, times, dollar amounts, etc. You can then use the recording capabilities provided in Script Builder to record your own speech for phrases unique to your application.

#### **⇒ NOTE:**

The quality of speech recorded through Script Builder is not as high, for example, as professionally recorded speech.

For detailed information about producing speech with Script Builder, see Chapter 10, "Speech Administration," of *Intuity CONVERSANT Voice Information System Version 5.0 Script Builder*, 585-310-727.

---

## **Purchasing the Standard Speech Package**

You can purchase the professionally recorded Standard Speech Package from AT&T. The Standard Speech Package provides the following:

- Most commonly used words and phrases, including the letters of the alphabet, pronounced and recorded in the three different inflections
- Essential numbers (“zero” through “twenty,” “thirty,” “forty,” “fifty,” “sixty,” “seventy,” “eighty,” “ninety,” “hundred,” “thousand,” and “million”)
- Days of the week
- Months of the year
- Ordinal numbers through 31 (“1st,” “2nd,” “3rd” through “31st”)
- Words “dollars” and “cents”

The Standard Speech Package is used to speak information using a variety of built-in speech formats. For example, if you want the system to speak a number using a money format, you might use number phrases followed by the phrase “dollars and,” then the number of cents and the phrase “cents.” For a complete listing of the standard speech phrases available from AT&T, see Appendix A, “Standard Speech Phrases”.

### **⇒ NOTE:**

Contact your AT&T representative if you are interested in purchasing the Standard Speech Package.

## **Purchasing an AT&T Custom Speech Package**

You can purchase a professionally recorded custom speech package from AT&T. You write out the script and AT&T records and digitizes the speech. Custom speech packages are available with both male and female voices.

Custom speech contains phrases designed specifically for the application you are developing. For example, “Thank you for calling AT&T,” is a custom speech phrase.

## **Advantages of AT&T Custom Speech**

The main advantage of purchasing a custom speech package from AT&T is that the speakers who record the custom speech phrases are the same speakers who record the standard speech phrases; therefore, a continuity exists among scripts that use both custom and standard speech.

### **⇒ NOTE:**

Contact your AT&T representative if you are interested in purchasing a custom speech package.

## Using Text-to-Speech

---

Text-to-Speech (TTS) is an advanced option that eliminates the need for recording speech. You enter the phrases to be spoken, and TTS synthesizes the speech. Basically, TTS converts text to speech in the following manner:

- The text is filtered to identify the sentence and phrase boundaries, expand conventional abbreviations, and translate non-alphabetic characters (for example, \$5 is translated to “five dollars”).
- Each word is labeled according to the part it plays in the text (noun, verb, preposition, etc.)
- The text is analyzed to determine pronunciation and emphasis.
- The text is further analyzed to determine timing and pitch, which is then associated with the pronunciation analysis.
- Synthesis converts the analyzed text into speech.

The TTS technology involves constructing speech by concatenating units of speech. When constructing speech from these units, the TTS feature adjusts parameters, such as pitch and duration, to make the outcome sound natural. Text filtering is critical because it:

- Expands abbreviations appropriate to the context (for example, “Dr.” could be expanded to “doctor” or “drive”, depending on the context)
- Adjusts for inappropriate punctuation (for example, “Dr” with or without a period is interpreted in the same way)
- Identifies names and addresses for special handling

TTS functionality is supported through the Intuity CONVERSANT Script Builder menu interface as well as with a TSM script instruction. The talk-off function and other VIS features for voice response work with TTS as they work with speech files. TTS also allows an application developer to use both pre-recorded phrases and TTS in the same application.

With some TTS applications, you may need to further customize the use of synthesized speech — for example, by adding silence delays, changing the speaking rate, or marking text as members of a more specific text category. While you can alter the manner in which TTS speaks a phrase, some applications require that you use escape sequences placed before, within, or after the spoken text. Refer to Appendix C, “Advanced Text-to-Speech Features,” for several ways to alter the manner in which TTS speaks a phrase. See *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for more information on using Text-to-Speech in your Script Builder application. See *Intuity CONVERSANT VIS V5.0 Application Development*, 585-310-227, for more information on using Text-to-Speech in your TSM application. *Intuity CONVERSANT VIS V5.0 IRAPI Programming Guide*, 585-310-226, for more information on using Text-to-Speech in your IRAPI application.

## **Using the Graphical Speech Editor**

---

The GSE enables you to record and edit speech files for applications to be used in the Intuity CONVERSANT VIS environment.

The following GSE options allow you to create and edit speech files designed to be run on VIS applications:

- Cut and Paste
- Copy and Paste
- Record
- Output Volume

You can also use the GSE to edit speech files that have been created with Script Builder. New phrases can be added to these prerecorded files and existing phrases can be changed, trimmed, or deleted.

## **Sharing Speech**

---

Sharing speech involves two or more applications that use the same speech, only one copy of which exists on your hard disk. If you have more than one application on a system, it will probably be more convenient for you to use the shared speech feature. This feature allows two or more applications to share common speech phrases. Performance advantages to sharing speech among applications are:

- Shared speech phrases need to be administered and recorded only once
- Shared speech phrases need to be stored only once, allowing disk space to be conserved

For more information on how to share speech in your Script Builder application, refer to Chapter 10, "Speech Administration," in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727.

## Importing Speech

You can import speech phrases from other applications and edit them with the GSE or Script Builder for use in your application. For example, suppose the following phrase was already recorded for another application:

“Thank you, please call Bank America again.”

You could import the phrase and edit it using GSE to say “Thank you, please call again,” and include it in your application. If you just wanted the “thank you” part of the phrase, you could use Script Builder to edit it. Script Builder can edit the beginning and ending of a speech phrase, whereas the GSE can edit any part of your speech phrase.

Another example of a phrase already recorded for another application is as follows:

For account balance, press 1.”

You could import the phrase and edit it using Script Builder to say “Press 1.”

For information on how to import speech into your Script Builder application, refer to Chapter 10, “Speech Administration,” in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727. For more information about editing speech, refer to Chapter 3, “Editing Speech”.

 **NOTE:**

With importing, two copies of the speech exist on the hard disk.

## **Recording Speech**

---

Depending on the speech development method you choose, the speech may need to be recorded. The following methods involve you recording speech or having the speech recorded for you:

- Using a professional speaker
- Purchasing a custom speech package from AT&T
- Producing self-recorded custom speech with Script Builder
- Producing self-recorded custom speech with the GSE

### **Using a Professional Speaker**

---

This section discusses what is involved in planning a recording session with a professional speaker and recording the speech on magnetic tape.

#### **Environmental Conditions**

A studio specifically designed for recording sessions is necessary when using a professional speaker. It should be noise-free and environmentally-controlled. Arrange for the recording environment to be quiet and acoustically “dead.” The room should also be carpeted with soft walls (drapes, carpet, etc).

#### **Equipment and Recording Parameters**

The recommended equipment includes a reel-to-reel tape recorder or high quality cassette player and amplifier. The reel-to-reel tape should be recorded at 7½ inches/second (19 centimeters/second). Post-processing such as filtering is not required. A video cassette recorder (VCR) with a digital audio processor also produces a high-quality recording.

#### **Equipment Specifications**

The recording apparatus and medium should provide the following:

- Dynamic range of at least 50dB
- Bandwidth from 100 to 8000Hz
- Flat frequency response in bandwidth
- Low noise insertion

## **Recording the Spoken Phrases on Magnetic Tape**

After the recording session has been planned and the voice script has been created, the professional speaker records the speech on magnetic tape. Refer to the sections earlier in this chapter called "Planning the Voice Script" and "Writing the Voice Script" for tips and information on planning and writing your voice script.

### **Recording Natural Sounding Speech**

The speaker uses the script to record entire sentences on tape so that the speech sounds natural. For example, for a temperature service, the following sentence can be recorded, although only parts of this sentence will be used:

"The current temperature is" sixty-seven "degrees Fahrenheit."

When this preceding sentence is encoded, the phrase "The current temperature is" can be encoded as one phrase and "degrees Fahrenheit" can be encoded as a second phrase. The speech phrase "sixty-seven" should be removed because "sixty-seven" is a combination of two phrases that are recorded separately and concatenated later (numbers and alpha characters are recorded as separate phrases). Refer to the section, "Recording Alpha and Numeric Characters with Inflections" later in this chapter.

A stock service may use a sentence similar to the following:

"The Dow was at" "eighteen" "forty" "three," "up" "7"  
"at the close of trading."

This entire sentence is recorded, but the sentence can be encoded as seven separate speech files that are concatenated later. The seven phrases which are encoded separately are shown in quotation marks (" ").

### **Recording Sets of Related Words**

When recording sets of related words, such as the days of the week, ordinal numbers, or the months of the year, use a frame sentence in a typical context. A frame sentence for the days of the week might be as follows:

The movie for "[the name of day]" is \_\_\_\_.



#### **NOTE:**

Remember that the AT&T Standard Speech Package includes days of the week, ordinal numbers, and months of the year. Therefore, it is not necessary to record these words if you purchase the Standard Speech Package. Refer to Appendix A, "Standard Speech Phrases", for a complete list of all phrases included in the Standard Speech Package.

During speech editing, the frame words preceding and following the day of the week are deleted and only the phrase that is inserted in place of “[the name of day]” is saved as a phrase.

The speaker, the studio manager, and a coordinator are usually present at the recording session. The customer for whom the speech is recorded may also be present. During the recording session, these individuals can provide feedback about the necessary inflections for words and phrases and the overall quality of the speech. Refer to Chapter 3, "Editing Speech", for more information on editing speech.

### **Recording Alpha and Numeric Characters with Inflections**

Record alpha and numeric characters with frame words that separate instances of initial, medial, and rising inflections for each letter and number.

**⇒ NOTE:**

Monitor the speaker during this phase of the recording session to insure that proper inflection is used and that volume and rhythm are constant.

**⇒ NOTE:**

Remember that the AT&T Standard Speech Package includes letters and numbers. Refer to Appendix A, "Standard Speech Phrases", for a complete list of all phrases included.

## Recording Speech with Script Builder

You can record speech directly into the VIS using Script Builder. See Chapter 10, "Speech Administration," of *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for instructions on recording speech in Script Builder.

## Purchasing a Custom Speech Package from AT&T

AT&T can provide you with professionally recorded custom speech through its speech recording service. The following lists the items AT&T needs to record custom speech phrases:

- A complete list of the phrase tags of all phrases that need to be recorded. The easiest way to do this is to use the Script Builder (F5) (LIST) function to generate a paper copy of all phrase tags in the application (unrecorded phrases are marked with an asterisk). Refer to Figure 2-1 for an example:

```
Custom Phrase Tags List
NOTE : a "*" in front of the phrase name implies that
       it is not recorded.
Follow all tt entries with a pound sign      spec_sut1
Follow all tt entries with a pound sign      sut
```

---

**Figure 2-1. Listing Custom Phrase Tags in Script Builder**

- The complete text of all the phrases that need to be recorded. Phrases must be written the exact way they are to be spoken.
- The phrase text should clearly indicate the phrase tag that corresponds to each phrase.
- Specify what type of speech coding should be used. Script Builder uses ADPCM (Adaptive Delta Pulse Code Modulation), PCM (Pulse Code Modulation), and SBC (Sub Band Coding).
- Specify whether the phrases are to be recorded using a male or female voice.

Contact the Speech Coordinator for the Intuity CONVERSANT VIS at 614-860-2260 for additional information on the custom speech recording service.

## **Digitizing Speech Phrases**

---

Recorded speech phrases are input from a reel-to-reel tape recorder, amplifier and microphone, or from a person over a telephone line. Once the speech is recorded, the speech must be digitized. Speech is digitized by encoding the speech in an acceptable format. Once the phrases are digitized, the phrases are stored as digital data.

You can digitize speech with the methods below:

- Digitize speech with the Graphical Speech Editor

If you record speech using the GSE, it is digitized automatically. Refer to the section earlier in this chapter for information about recording speech with the GSE.

- Digitize speech with Script Builder

If you have recorded the speech using Script Builder, it is digitized automatically. For more information about Script Builder's speech administration capabilities, refer to Chapter 10, "Producing Speech," of *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727.

- Use AT&T to digitize the speech

If you want AT&T to digitize the speech, contact your AT&T representative for information about where to send the recorded speech.

## **Installing Speech**

---

After the speech has been digitized, you need to install the speech onto your system.

### **Installing a Script Builder or TSM Application**

---

If your application has been created in Script Builder, refer to Chapter 10, "Speech Administration," in the section "Restoring Speech" in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for specific information on how to install speech onto your system.

### **Installing Speech in an IRAPI Application**

---

If your application has been created using the C language, perform the following procedure to install speech:

1. Insert the floppy disk into the drive.
2. Go to the appropriate speech filesystem. For example, if you are using the default speech filesystem, enter the following:

```
cd /home2/vfs/talkfiles
```

3. Enter **mkdir <talkfile #>**

Make sure that you select a talkfile number not already in use.

For example, if you are using talkfile 100, you would enter the following:

```
mkdir 100
```

4. Copy the speech phrases to the directory specified above.

When the system prompt is displayed, the speech is loaded onto the hard disk and can be accessed by applications.

#### **⇒ NOTE:**

If you are adding speech to an existing application, be aware that the system overwrites any existing speech files that have the same name as a file being added.

With an IRAPI application, you can put speech anywhere on your system. For information on accessing speech files in an IRAPI application, see Chapter 3, "IRAPI Run-Time Services," in *Intuity CONVERSANT VIS V5.0 IRAPI Programming Guide*, 585-310-226.

**What's in This Chapter**

This chapter includes the following information:

- Using the Graphical Speech Editor (GSE) to edit speech files
- Using Script Builder to edit speech files
- Using third-party speech-editing systems to edit speech files

## **Introduction to the Graphical Speech Editor**

---

The GSE is a X-Windows based graphical user interface (GUI) that can be accessed and operated with a three-button mouse and a keyboard.

The GSE is able to play and record speech, and can accept audio speech signals from either a microphone, a cassette tape from audio line input, or a telephone unit, supported through one of the standard Tip/Ring (T/R) circuit cards. These speech signals are digitally stored under appropriate file names for use by VIS applications. The contents of these files can be played back and heard via a telephone handset or a speaker.

### **Making Changes with GSE**

---

GSE provides a simple way of making changes to existing speech phrases by allowing you to cut, copy, and paste speech segments in either a speech file or across multiple speech files. It also provides the ability to change the volume of individual speech segments.

### **Supported Speech Formats**

The GSE package supports the editing and manipulation of speech recorded in the following formats:

- 64kbps Pulse Code Modulation (PCM) (Mu-law)
- 32kbps Adaptive Differential Pulse Code Modulation (ADPCM) (Mu-law)
- 16kbps ADPCM (Mu-law)

The GSE provides the ability to convert speech files between Mu-law PCM formats and user-selected ADPCM formats. To minimize degradations due to multiple conversions of speech, all speech storage and intermediate manipulations are performed using 64kbps PCM. The GSE buffer is capable of handling up to four minutes of speech in its buffer, whether the speech consists of one phrase or a series of phrases.

### **Working with Segments of Speech**

---

The GSE feature allows you to display a specific segment of speech that appears as a graphical waveform on your system monitor. You mark an area by moving your mouse over a segment of speech to highlight the speech (waveform) that is needed. This highlighted region can then be cut or copied to a buffer, and subsequently pasted to some other location in the waveform. The waveform displayed on the monitor is the most current result of all editing operations you performed on a particular speech file.

---

## Accessing the Graphical Speech Editor

The GSE has a user-friendly graphical interface for creating and editing speech files. This interface includes windows that can be accessed to perform different functions associated with the retrieval, creation, editing, and saving of speech files. This section also describes installing and administering the mouse, mouse operations, and how to use the keyboard to perform editing functions.

### Installing and Administering the Mouse

For instructions on installing the mouse, refer to the Intuity CONVERSANT VIS Version 5.0 Hardware Installation book specific for your platform. For instructions on administering the mouse, refer to *UnixWare System Administration System Setup and Configuration*, 585-350-908.

**CAUTION:**

When referring to Chapter 1, “Setting Up the Work Environment,” in *UnixWare System Administration System Setup and Configuration*, 585-350-908, in the section “Adding a Mouse Device,” note the following:

In order to save the changes to the mouse configuration that you have just added, you need to select **U** (Update Mouse Configuration and Quit) and then enter **mouseadmin** before you can test the mouse configuration by selecting **T** (Test your mouse configuration). The procedure is shown in the reverse order in Chapter 1, “Setting Up the Work Environment.”

### Installing the Graphical Speech Editor

The procedure for installing the Graphical Speech Editor is described in the *Intuity CONVERSANT VIS V5.0 Software Installation*, 585-310-151. Refer to the Intuity CONVERSANT VIS Version 5.0 Hardware Installation book specific for your platform for information about GSE hardware configurations.

**NOTE:**

The Intuity CONVERSANT VIS must be stopped while using the Graphical Speech Editor. Entering the **gse** command results in a system prompt asking you if it is okay to stop the VIS.

## Initiating the Graphical Speech Editor

To initiate the Graphical Speech Editor, enter the following command:

**`gse [-l<chan#>] [-p<playchan#>] [-r<recchan#>]`**

where:

- l = listen channel (input and output)
- p = play channel (output only)
- r = record channel (input only)

The designated channels must be configured for both input and output; this facilitates recording and playing speech phrases. The -l channel is used for telephone lines and incorporates both audio input and output. Alternatively, the -p and -r channels must be used together to establish the audio input/output, where -r (input) is designated for the microphone, and -p (output) for the speaker.

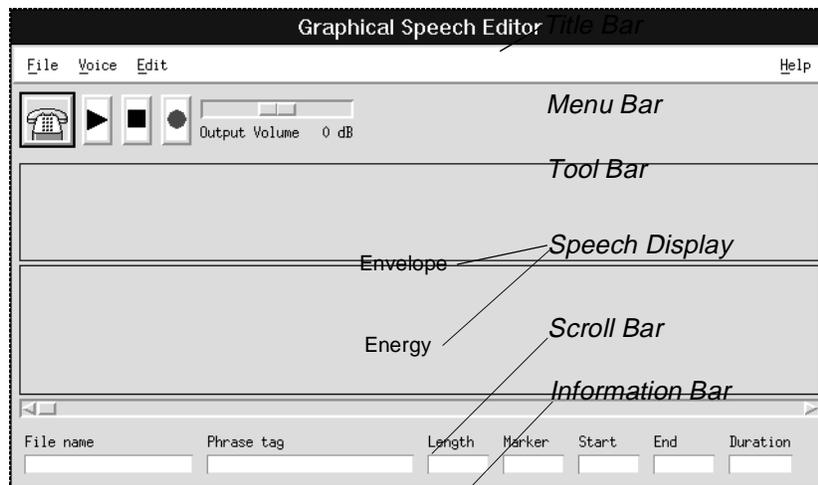
The Graphical Speech Editor user interface appears as shown in Figure 3-1.

For more information about the **gse** command, see *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230.



**CAUTION:**

*GSE channel numbers are not the same as VIS channel numbers. Refer to Appendix C, "Computing Channel Numbers" for an explanation of how to determine channel numbers for the l, p, and r channels.*



**Figure 3-1. Graphical Speech Editor Window**

---

## **Graphical Speech Editor User Interface**

This section describes the layout and operation of the GSE user interface. The GSE user interface enables you to record and edit speech files for applications to be used in the Intuity CONVERSANT VIS environment. The following components comprise the GSE user interface:

- "Title Bar"
- "Menu Bar"
- "Tool Bar"
- "Speech Display Area"
  - Envelope
  - Energy
- "Scroll Bar"
- "Information Bar"

Figure 3-1 illustrates all of these components.

### **Title Bar**

---

The title bar shown in Figure 3-2 expands across the top of the Graphical Speech Editor window to identify the graphical user interface feature of the Intuity CONVERSANT VIS.



---

**Figure 3-2. Title Bar**

## Menu Bar

---

The menu bar shown in Figure 3-3 is comprised of the following subfunctions:

- File
- Voice
- Edit
- Help



---

**Figure 3-3. Menu Bar**

## File Menu

The File menu offers the following choices:

- Open  
To access an existing speech file, select Open from the File menu. Note that you must first create and save new files before you can open them.
- Save  
After you have edited a speech file, save your changes by selecting Save from the File menu.
- Save As  
You must name and save new speech files. To do this, select Save As from the File menu.
- Exit (Ctrl+Q)  
The Exit option closes the GSE and returns you to the system prompt. You can also exit the editor at any time by pressing **Ctrl** **Q** simultaneously.

## Voice Menu

The Voice menu offers the following choices:

- Hook

The telephone receiver (handset) can either be *on-hook* or *off-hook*. To play or record speech, the receiver must be off-hook.

When you select this item from the menu, a submenu is displayed from which you can identify either of the two receiver states.

- Play

After a speech file has been opened and the telephone receiver is off-hook, select Play from the Voice menu to hear the message contained in the speech file.

- Stop

Activate the Stop function to end a recording session or to discontinue a message playback.

- Record

Select the Record function to start a recording session. Typically, you would select Record after the telephone receiver is off-hook.



**NOTE:**

You can also select all Voice menu items from their corresponding Tool menu icons.

## Edit Menu

The Edit menu offers the following choices:

- Cut

This editing function enables you to remove a selected segment of speech from the speech file. You may want to cut a segment of speech either to delete it entirely from the speech file, or to paste it at a different location.

- Copy

You can copy segments of speech into a buffer for pasting at different locations in the speech file. You must select the segment before you can implement the Copy function.

- Paste

You can paste segments of speech that have either been cut or copied into the buffer anywhere in the speech file. The selected segment of speech is pasted at the present location of the marker.

Use the middle-button of the mouse to move the marker across the speech display.

- Volume

Select this editing function to set the volume at the level the speech file is to be heard by the caller.

- Undo

While editing a speech file, use this function to undo the last change made. The Undo function applies to all the editing features offered by the GSE.

 **NOTE:**

Instead of clicking the mouse to perform any of the above listed editing functions, you can alternatively use the corresponding key sequences.

## Help Menu

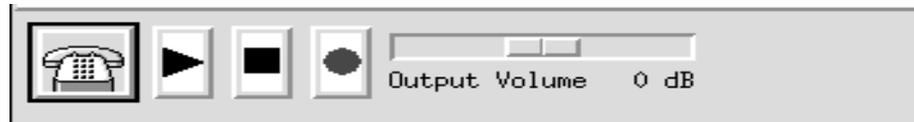
The Help menu is reserved for future use.

---

## Tool Bar

---

The tool bar (Figure 3-4) is comprised of five icons that you can use to perform the basic editing functions offered by GSE. Each of these icons represents a function that you can also perform via appropriate selections on the menu bar. These functions are described below.



**Figure 3-4. Tool Bar**

- Telephone Icon

This icon reflects the *on-hook* or *off-hook* state of the telephone handset. To place the telephone in an off-hook state (Figure 3-5), click on the telephone icon.



**NOTE:**

The telephone handset must be off-hook prior to recording speech or listening to a playback via the Play function.

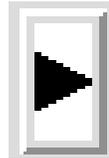


**Figure 3-5. Telephone Icon Showing the Off-Hook State**

- Play Icon

Activate the play icon (Figure 3-6) to listen to a playback of recorded speech.

---



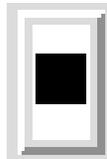
---

**Figure 3-6. Play Icon**

- Stop Icon

Activate the stop icon (Figure 3-7) to stop recording or playing speech.

---



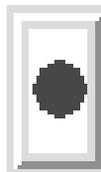
---

**Figure 3-7. Stop Icon**

- Record Icon

Activate the record icon (Figure 3-8) to start recording speech.

---



---

**Figure 3-8. Record Icon**

---

- Output Volume Icon

You can adjust the volume at which a speech file is to be played by dragging the volume button (Figure 3-9) in either direction with the mouse. The increased or decreased output level, measured in decibels, is reflected below the icon.

**NOTE:**

Changing the volume in this manner only affects the speech being played during a GSE session. To permanently change the volume, you need to change it through the Edit menu.



---

**Figure 3-9. Volume Icon**

## Speech Display Area

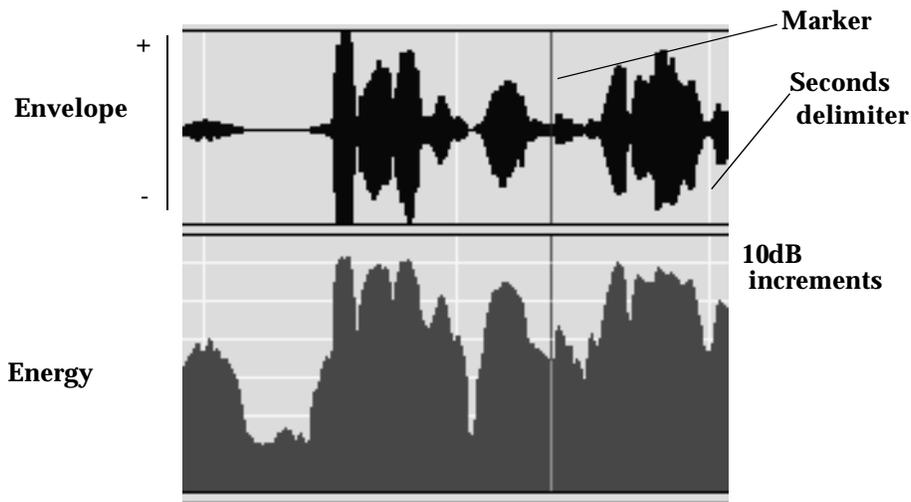
---

The speech display area represents a graphical image of speech. Approximately seven seconds of speech, from a speech file that can be up to four minutes long, are displayed at a time.

Two windows concurrently display separate graphical representations of the same digitized speech file (Figure 3-10). The envelope and energy representations of the audio signal are essentially the same. Their difference lies in their respective interpretative values to you.

Envelope is a linear scaled representation of voltage on a line. It shows the sound wave amplitude at different intervals of time. The bubble-like structures in the envelope window (Figure 3-10) are graphical representations of an audio signal oscillating between the positive and negative extremes.

Literally translated, energy is the output level of sound behind every phonetic utterance of speech. This energy is vertically measured in 10-dB increments and graphically plotted on the display. The energy window displays the amount of energy in the audio signal.



---

**Figure 3-10. Graphical Speech Editor Speech Display**

---

## Scroll Bar

---

The scroll bar shown in Figure 3-11 enables you to pace through the contents of the speech file, by clicking on the end arrows in either direction. Using the middle button on the mouse accelerates this process.

A rectangular column within the scroll bar identifies the location of the displayed segment in relation to the total contents of the speech file. The width of this column reflects the proportion of the speech file being displayed.

You can also drag this rectangular column to the approximate location of the speech file you want to access.



---

**Figure 3-11. Scroll Bar**

---

## Information Bar

---

The information bar (Figure 3-12) is located at the bottom of the screen. It provides information pertinent to GSE operations, such as the speech file name, phrase tag, length of speech, etc.

---

File name	Phrase tag	Length	Marker	Start	End	Duration
/usr/dsp/a.brutus	welcome to brutus	5,58	0,00	0,00	0,00	0,00

---

**Figure 3-12. Information Bar**

The information bar is comprised of the following windows:

- File Name Window

The File name window identifies the name of the speech file.

- Phrase Tag Window

The Phrase tag window is used for applications developed on Script Builder. The phrase tag is for identification purposes only. Refer to Chapter 1, "Overview of Speech on the Intuity CONVERSANT VIS", for information about phrase tags.

- Length Window  
The Length window identifies the entire length of the speech file in hundredths of a second.
- Marker Window  
The Marker window identifies the position of the marker (vertical cursor) while speech is played in hundredths of a second.
- Start Window  
The Start window identifies the starting point of a segment of speech that has been selected for editing in hundredths of a second.
- End Window  
The End window identifies the end boundary of a segment of speech that has been selected for editing in hundredths of a second.
- Duration Window  
The Duration window identifies the duration of the selected segment, or the difference between the start and end limits of the segment in hundredths of a second.

## **Editing a Speech File with GSE**

---

This section provides a detailed description of how to use the different editing functions provided with the GSE. The following issues are discussed:

- "Using the Mouse"
- "Opening a File"
- "Setting the Marker"
- "Selecting a Segment of Speech"
- "Cutting and Pasting Speech"
- "Copying and Pasting Speech"
- "Adjusting the Volume"
- "Undoing Changes"
- "Saving Changes"
- "Exiting"

### **Using the Mouse**

---

This section summarizes how to use the mouse:

- Use the left button for:
  - Selecting menu items
  - Selecting a segment of speech
  - Modifying the length of the selected segment
  - Moving horizontally in either direction with the scroll bar
  - Moving vertically up or down with the scroll bar (opening files)
- Use the middle button for:
  - Moving the marker
  - Setting a position on the scroll bar



**NOTE:**

The right button has been reserved for future use.

## Opening a File

---

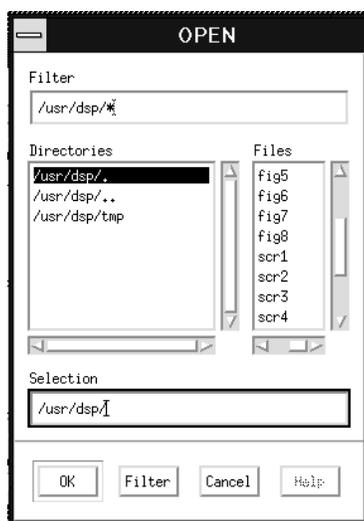
Before you can edit an existing speech file, you must open it or bring it into the GSE.

To open a file, perform the following procedure:

1. Select File from the menu bar.
2. Click on Open with the left button on your mouse.

The Open window (Figure 3-13) appears.

---



---

**Figure 3-13. Open File Screen**

This window provides you with scrolling tools that facilitate the identification of files within a specific directory. In addition, the Filter button expands the identification of a path, or pattern, from which a particular speech file can be retrieved.

3. Click on the OK button to open the file or press the cancel button to return to the speech-editing screen.

---

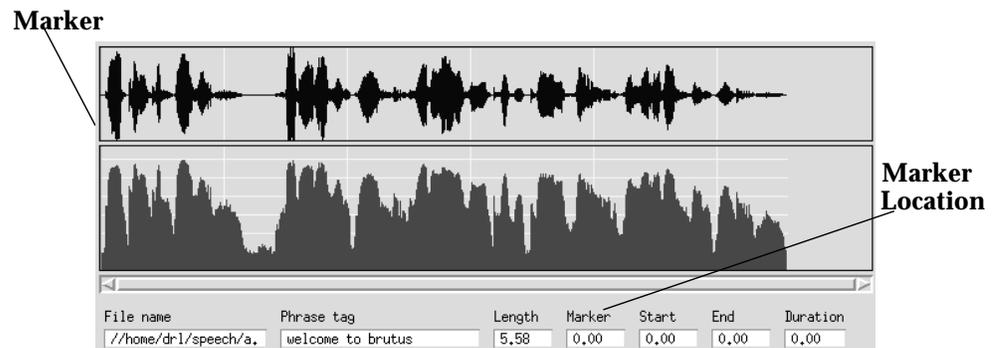
## Setting the Marker

---

The Marker is a light blue vertical line that identifies a particular segment of speech on the speech display. This segment is reflected in the Marker section of the information bar at the bottom of the screen (Figure 3-14).

To move the Marker, press the middle button on the mouse. Release the mouse button to position the Marker at a specific location.

---



---

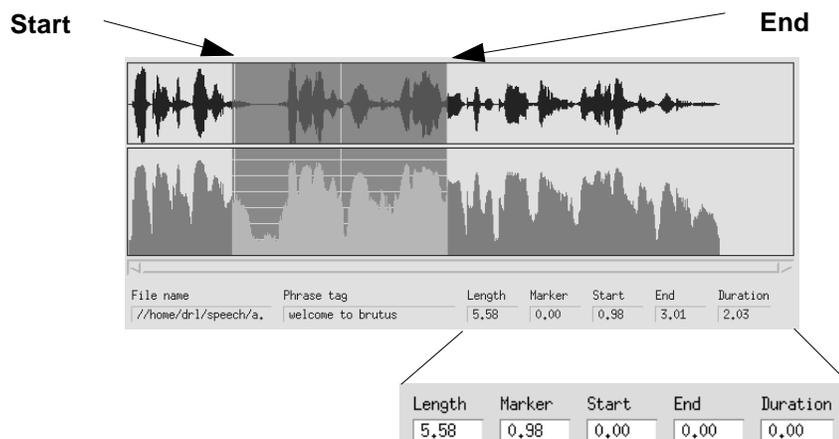
**Figure 3-14. Speech Marker**

## Selecting a Segment of Speech

To select a segment of speech, perform the following procedure from the speech display area:

1. Press the left mouse button to identify the start position, drag the marker to the desired location, and release the marker to mark the end position.

The selected segment is then highlighted within the identified starting and ending boundaries (Figure 3-15).



**Figure 3-15. Selecting Segments of Speech**

If you look at the information bar at the bottom of the screen, you will notice the Start and End delimiters of the selected segment presented in hundredths of a second (Figure 3-15). The Duration window identifies the total time that the selected segment of speech stretches across.

After you have selected a segment of speech, you can modify its length without reselecting it.

2. Press the left button of the mouse at either end point and drag it to the desired location to modify the length.
3. Use the left button of the mouse as described in the preceding paragraph to lengthen the selected segment one window at a time.

---

## Listening to Speech

---

To listen to a segment of speech, perform the following procedure:

1. Open a speech file. See the previous section, "Opening a File".
2. Make sure the telephone receiver is off-hook.
3. Select a segment of speech. See the previous section, "Selecting a Segment of Speech".
4. Select Play from the Voice menu to hear the speech file.
5. Click on the OK button.

### NOTE:

Once you click on OK, the Graphical Speech Editor looks for the presence of a voice header. If you use the **`gse_copy`** or **`gse_copypl`** command, this voice header is added automatically to your speech file. Otherwise, you need to make sure that the speech file is in the correct format and then add a voice header with the **`addhdr`** command. If the speech file does not have a voice header, you receive the following message:

FILE NOT A SPEECH FILE

Refer to the section later in this chapter, "Converting Speech Files", for more information about preparing your speech files for use with the Graphical Speech Editor. See *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230, for information about the commands mentioned above.

---

## Cutting and Pasting Speech

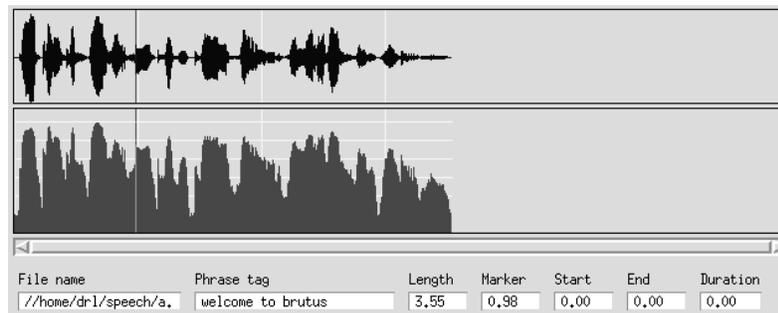
---

A selected segment of speech can be removed from the speech file by using the Cut function. You can then paste the speech into a different part of the file with the Paste function. The Cut function removes the segment of speech from the file and simultaneously copies it into the buffer. This enables you to later paste the segment.

To remove a segment of speech from a speech file and then paste the speech into a different part of the file, perform the following procedure:

1. Select the segment of speech to be removed from its present location. Refer to the instructions listed in the previous section, "Selecting a Segment of Speech".
2. Select Edit from the menu bar.
3. Select Cut or press **`Ctrl`** **`T`**.

The selected segment of speech is removed from the file. In Figure 3-16, the segment of speech selected in Figure 3-15 is now removed from the speech file.



---

**Figure 3-16. Cutting Segments of Speech**

4. Move the marker to the destination area by using the middle button of the mouse.
5. Select Edit from the menu bar.
6. Select Paste, or press **Ctrl** **P**.

The contents of the system buffer are pasted at the identified location of the speech file.

**NOTE:**

The above process can be repeated as long as the speech file does not exceed 4 minutes.

---

## Copying and Pasting Speech

---

Use the copy function to copy selected segments of speech to a different location of the speech file. This function copies the selected segment into the buffer without cutting it from its original position.

 **NOTE:**

You must use the Copy function before the Paste function.

To copy a segment of speech, perform the following procedure:

1. Select the segment of speech to be copied.
2. Click on the Copy function with the mouse or press **Ctrl** **C** simultaneously.

The system copies the selected segment of speech into the system buffer. You can now paste the contents of the buffer into different parts of the speech file.

3. Move the marker to the destination area by using the middle button of the mouse.
4. Select Edit from the menu bar.
5. Select Paste.
6. Press **Ctrl** **P** simultaneously.

The contents of the system buffer are pasted at the identified location of the speech file.

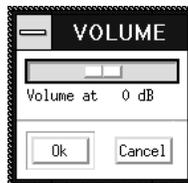
## Adjusting the Volume

---

Use the Volume function to adjust the volume at which a caller hears a speech file.

To adjust the volume of speech, perform the following procedure:

1. Select the segment of speech.
2. Access the Edit menu from the menu bar.
3. Click on Volume (Figure 3-17), or press **Ctrl** **V** simultaneously.
4. Use the left mouse button to drag the volume button in either direction.  
The increased or decreased output level, measured in decibels, is reflected below the icon.
5. Click on the Ok button to save the volume setting.
6. Select Cancel to return to the speech-editing screen.



---

**Figure 3-17. Modifying the Speech Volume**

---

## Undoing Changes

---

You can undo the last change you made to a speech file when performing any of the preceding functions.

To undo the last change, perform the following procedure:

1. Access the Edit menu from the menu bar.
2. Click on the Undo function with the mouse or press **Ctrl** **U**.

---

## Saving Changes

---

After you have edited a speech file, save the changes by selecting Save from the File menu. To name and save new speech files, perform the following procedure:

1. Select File from the menu bar.
2. Select Save As.

The Save As screen appears (Figure 3-18). This screen provides scrolling tools that facilitate the naming of files within a specific directory. The Filter button enables identification of a path or pattern to which a particular speech file is to be saved.

3. Click on the OK button to complete the save, or click on the Cancel button to return to the speech-editing screen without saving the changes.



**Figure 3-18. Save As... Screen**

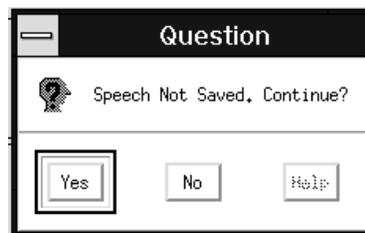
## Exiting

---

The Exit selection from the File menu closes the GSE and returns you to the system prompt. You can also exit the editor at any time by pressing **Ctrl Q**. Once you exit the Graphical Speech Editor, click and hold down the left mouse button and simultaneously press **Ctrl** to exit X windows.

**⇒ NOTE:**

If you have been working on a speech file and choose to exit prior to saving the file, a window with the message shown in Figure 3-19 appears.



---

**Figure 3-19. Exiting without Saving the Speech File**

---

## Converting Speech Files

---

This section describes the utilities provided with Intuity CONVERSANT VIS to integrate the GSE into the application development environment. These integration utilities are designed to perform the following functions:

- Convert speech stored in the VIS speech filesystem for use with the GSE
- Convert GSE output files and store their contents in the VIS speech filesystem

There are two methods of retrieving speech files between the GSE and VIS. One pertains to existing speech pools that support Script Builder applications. The other pertains to those phrases that were created for the application. These phrases are stored in system talkfiles.

The **gse\_copypl** command copies phrases from an identified speech pool into UNIX files which are in the PCM64 format (required by the GSE). Conversely, the **gse\_addpl** command adds (restores) phrases to a specific speech pool from correspondingly named UNIX files in the GSE format.

For single phrases, the **gse\_copy** command, copies the phrase (identified by *talkfile* and *phrase number*) into a UNIX file. Conversely, the **gse\_add** command adds (restores) the phrase to *the talkfile* (identified by the same *talkfile* and *phrase number*).

The **gse\_copypl** command uses the **gse\_copy** routine to extract individual phrases. Similarly, the **gse\_addpl** command uses the **gse\_add** routine to restore individual phrases.

The following sections detail the process of retrieving speech from speech pools, converting it to a form that can be edited by the GSE, and reversing the entire process.

## Retrieving Files from Speech Pools

---

This section includes information on how to extract speech files from speech pools and prepare the files for editing by the GSE. The VIS must be running to extract speech-files for editing.

To retrieve speech files from a speech pool and prepare the files for editing with GSE, enter the following command:

```
gse_copypl <speech pool> <output dir> [<file1>... <fileN>]
```

where *<speech pool>* is the name of the speech pool from which the speech is to be retrieved; *<output dir>* is the name of the directory where the output files are to be put; *<file1> <file2>... <fileN>* are optional file names identifying the particular phrase names to be extracted.

The **gse\_copypl** command reads the phrase listfile belonging to the speech pool (**/speech/talk/<speech pool>.pl**) to determine the talkfile, phrase numbers, and output file names for the phrases to be extracted. You may determine which speech pools are being used by an application by examining the Shared Speech pools parameter of the Script Builder application. Refer to Chapter 7, "Defining Parameters," in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for more information about the Shared Speech Parameter in Script Builder.

Output files are placed in the directory **<output dir>**. If no output file names are specified on the command line, all phrases in the speech pool are extracted. If file names are given, only the phrases with those file names that are specified in the phrase listfile are extracted.

### Example

Take the example of a phrase listfile named **talk3.pl**. The file is stored in the **/speech/talk** directory. The name of the speech pool represented is **talk3** (Table 3-1). The talkfile number being used by the speech pool (103) is on the first line of the file. Each subsequent line of the file contains a file name, phrase number, and phrase tag (in that order) for each phrase in the speech pool.

Therefore, typing the command, **gse\_copypl talk3 /speech/talk/talk3.files** would extract phrases 1000, 1001 and 1002 from talkfile 103 and put them in files **f1000**, **f1001** and **f1002** (respectively) in the directory **/speech/talk/talk3.files**. These files would then be ready for editing with the GSE.

Typing the command, **gse\_copypl talk3 /speech/talk/talk3.files f1001 f1002**, would however, only extract phrases 1001 and 1002 from the talkfile.

**Table 3-1. Phrase Listfile (talk3) Format**

File Name	Phrase Number	Phrase Tag
103		Phrase list for application: talk3
f1000	1000	This is the first Phrase Tag
f1001	1001	This is the second Phrase Tag
f1000	1002	This is the third Phrase Tag

### Restoring Files to Speech Pools

To restore speech files to their original format and add them to the appropriate speech pool, enter the following command:

```
gse_addpl <speech pool> <input dir> <codetype> [<file1>... <fileN>]
```

where *<speech pool>* is the name of the speech pool to which the speech is to be added; *<input dir>* is the name of the directory where the GSE edited files are located; *<codetype>* is either "pcm64", "adpcm16" or "adpcm32"; and, *<file1>* *<file2>*... *<fileN>* are optional file names identifying the phrase names to be added.

The **gse\_addpl** command reads the phrase listfile in the speech pool (**/speech/talk/<speech pool>.pl**) to determine the talkfile, phrase numbers and file names of the phrases to be added. You may determine which speech pools are being used by an application by examining the Shared Speech Pools parameter of the Script Builder application. If no file names are specified on the command line, all phrases in the speech pool for which files are found in **<input dir>** are added. If file names are given, only the phrases with the particular file names that are specified in the phrase listfile are added.

Script Builder designates an unrecorded phrase in the phrase listfile by using a negative phrase number. If **gse\_addpl** is used to add a previously unrecorded phrase, the phrase number is changed to its positive value to indicate that the phrase exists. Applications should then be verified and installed with the specific speech pool through Script Builder. Refer to Chapter 7, "Defining Parameters," in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, for more information about the Shared Speech Parameter in Script Builder.

### Example

Take the example of a phrase listfile named **talk3.pl**. The file is stored in the **/speech/talk** directory. The name of the speech pool it represents is **talk3** (Table 3-1). The talkfile number being used by the speech pool (103) is on the first line of the file. Each subsequent line of the file contains a file name, phrase number and phrase tag (in that order) for each phrase in the speech pool.

Therefore, entering the command

```
gse_addpl talk3 /speech/talk/talk3.files adpcm32
```

would add phrases 1000, 1001 and 1002 to talkfile 103 in ADPCM32 format from files f1000, f1001 and f1002 (respectively) in the directory **/speech/talk/talk3.files**. These files would not be removed by the **gse\_addpl** command.

Entering the command,

```
gse_addpl talk3 /speech/talk/talk3.files pcm64 f1001 f1002
```

would instead, add only phrases 1001 and 1002 from the talkfile in PCM64 format.

---

## Retrieving Files from the Speech Filesystem

This section includes information on how to retrieve speech files from the VIS speech filesystem and how to prepare the files for editing by the GSE. The VIS must be running to extract speech files for editing.

To retrieve speech files from the speech filesystem and prepare the files for editing with GSE, enter the following command:

```
gse_copy <talkfile> <phrase> <outputfile> [“<phrasetag>”]
```

where <talkfile> is the talkfile number; <phrase> is the phrase number; <outputfile> is the output file; <phrasetag> is an optional 50 character string that will be placed into the GSE voice header of the output file. The GSE displays the tag value when the file is being edited.

The **gse\_copy** command may be used in unique situations where a phrase not belonging to a speech pool needs to be edited. In such an event, you must know the talkfile and phrase numbers of the phrases to be edited.

If this information is not known, you may be able to determine them with the following commands:

- When no information is available, enter **list talkfile**

This command provides information on all phrases stored in the speech filesystem sorted by talkfile number. The needed talkfile and phrase numbers can be determined by searching for the name of the phrase (that is, the phrase tag).

- If the talkfile number is known, the following command can be used for a list of all phrases in the given talkfile: **list talkfile <tnum>** where, <tnum> is the talkfile number.
- If the phrase number is known, but not the talkfile number, the following command can be used for a list of all talkfiles in which the given phrase number is used: **list phrase <phnum>** where, <phnum> is the phrase number.

When the talkfile and phrase numbers are known, you may use the **gse\_copy** to retrieve the phrase for editing.

### Example

Entering the command

```
gse_copy 103 1000 /usr/speech/103/1000
```

would extract phrase 1000 from talkfile 103 and put it in the **/usr/speech/103/1000** file for editing by the GSE.

The **gse\_copy** command is implemented as a UNIX shell program which uses existing commands to perform all the needed tasks.

### NOTE:

You must keep track of which extracted files go with what talkfile and phrase in order to put the speech back in the same place after editing. It is suggested that *<file>* be the same as the phrase number and that the directory containing *<file>* be named the same as the talkfile number.

## Restoring Files to the Speech Filesystem

To restore speech files to their original speech file, enter the following command:

```
gse_add <talkfile> <phrase> <codetype> <inputfile>
```

where, <talkfile> is the talkfile number; <phrase> is the phrase number; <codetype> is either "pcm64", "adpcm16" or "adpcm32"; <inputfile> is the input file.

The **gse\_add** command may be used in situations when a phrase not belonging to a speech pool needs to be added. In that event, you must know the talkfile and phrase numbers of the phrases to be added. If these are not known, you may be able to determine them by using the **list** command in the manner described previously.

Once the talkfile and phrase numbers are known, you may use the **gse\_add** command to add the phrase to the talkfile.

### Example

Entering the command **gse\_add 103 1000 adpcm32 /usr/speech/103/1000** would add phrase 1000 to talkfile 103 in ADPCM32 format from the file **/usr/speech/103/1000**.

## **Using Script Builder for Editing**

---

Speech editing with Script Builder consists of trimming off pieces of the recording, from one or both ends, and playing the edited recording, until it is edited as desired. You can restore what you have trimmed in any given editing session, if too much is trimmed. For detailed information about editing speech files with Script Builder, refer to Chapter 10, "Speech Administration," in *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, in the section "Editing Speech."

---

## Using Third-Party Speech Editing Systems

---

Although AT&T offers the GSE feature as part of Intuity CONVERSANT VIS Version 5.0 and recommends its use, the Intuity CONVERSANT system's open architecture also enables the use of third-party speech-editing systems.

This section details the prescribed format for speech files to enable the use of a third-party editing system. It also describes how to facilitate file transfer.

A section on using the Audio Works Station with Intuity CONVERSANT VIS V5.0 is provided as an example. Audio Works Station is a third-party speech-editing system.

To use a third-party speech-editing system to edit VIS speech files, you will need to perform the following procedures:

1. "Retrieving Files from the Speech Filesystem".
2. "Converting Speech File Formats" from the VIS format to PCM 64kbps.
3. "Transferring Speech Files" from the VIS to the third-party editing system.
4. Edit the speech files within the third-party editing system.
5. Move the speech files back to the VIS system.
6. Convert the edited speech files to PCM 64kbps.
7. Install the speech files in the VIS speech filesystem

These procedures are described in the following sections as well as in the Audio Works Station example.

## Retrieving Speech Files from the Speech Filesystem

---

This section includes information on how to retrieve speech files from the VIS speech filesystem. The VIS must be running to extract speech files for editing.

To retrieve speech files from the speech filesystem, enter the following command:

```
gse_copy <talkfile> <phrase> <outputfile> [“<phrasetag>”]
```

where <talkfile> is the talkfile number; <phrase> is the phrase number; <outputfile> is the output file; <phrasetag> is an optional 50 character string that is placed into the GSE voice header of the output file. The GSE voice header needs to be stripped before using a third-party speech editing system. Use the **striphdr** command to strip the voice header. See *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230, for information about the **striphdr** and **gse\_copy** commands.

**⇒ NOTE:**

You do not have to purchase the Graphical Speech Editor in order to have access to the commands discussed above. These commands (**gse\_copy** and **striphdr**) are part of the standard Intuity CONVERSANT VIS Version 5.0 offer.

## Converting Speech File Formats

---

To use the VIS speech file with a third-party speech-editing system, you may need to convert speech files from one format to another.

**⇒ NOTE:**

VIS utilities automatically convert VIS Mu-law files into VIS ADPCM formats. VIS utilities do not facilitate Mu-law PCM conversion.

To convert speech encoded in the Mu-law PCM format to ADPCM 32 or ADPCM 16, the following procedure is necessary:

1. Convert the speech file encoded in u-law PCM format into a PCM format.
2. Convert the linear PCM format speech file to either ADPCM 32 or ADPCM 16 to Mu-law PCM format. For more information about speech-file formats, refer to Chapter 2, "Developing Speech".

**⇒ NOTE:**

It is recommended that you do not use the ADPCM16 format because it significantly decreases speech quality.

 **NOTE:**

A degradation in speech quality may occur whenever speech encoded in one format is converted to another format. Each time you repeat this conversion process, further degradation may occur.

## Transferring Speech Files

---

To transfer a speech phrase from an editing system to the VIS or vice versa, it may be necessary to move the speech file between the two systems. Two ways to transfer speech files exist and the method depends on your configuration.

- Modems via serial communication transfers over an interconnected local area network (LAN)
- Magnetic media such as tapes or floppy disks

### ⇒ NOTE:

You need to manage all speech phrases stored as UNIX files and keep track of the applications to which the speech files belong. You may need to understand the format of phrase-list files. These files map system speech phrases to talkfiles, and phrase tags to phrase numbers and file names.

To restore speech files to their original speech filesystem, enter the following command:

```
gse_add <talkfile> <phrase> <codetype> <inputfile>
```

where, *<talkfile>* is the talkfile number; *<phrase>* is the phrase number; *<codetype>* is either "pcm64", "adpcm16" or "adpcm32"; *<inputfile>* is the input file. You also need to add the GSE voice header back into the speech file. Use the **addhdr** command to do this. See *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230, for information about the **addhdr** command.

The **gse\_add** command may be used in situations when a phrase not belonging to a speech pool needs to be added. In that event, you must know the talkfile and phrase numbers of the phrases to be added. If these are not known, you may be able to determine them by using the **list** command in the manner described previously.

Once the talkfile and phrase numbers are known, you may use the **gse\_add** command to add the phrase to the talkfile.

The VIS also provides a utility for transferring UNIX system files from and to MS-DOS formatted files.

- The **doscpc** command allows you to copy a
  - UNIX file to a MS-DOS formatted file with the command  
**doscpc UNIX-file a: /MSDOS-file**
  - MS-DOS formatted file to a UNIX filesystem, with the command  
**doscpc a: /MSDOS-file UNIX-file**

The following is a list of commands that can be used with the GSE. For more information, see *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230.

- **gse\_copy**  
Extracts a speech phrase from the VIS speech filesystem to a UNIX file
- **gse\_add**  
Transfers a speech phrase from a UNIX file to the speech database
- **erase**  
Deletes phrases from the speech database
- **list**  
Lists the directory entries for specific phrases
- **gse\_addpl**  
Adds multiple files from the Script Builder application speech pool to the speech filesystem
- **gse\_copypl**  
Copies multiple speech phrases from a Script Builder application speech pool

## Using the Audio Works Station

---

The Audio Works Station is a speech-editing system offered by Bitworks Inc. It runs on the MS-DOS operating system platform. The product consists of an audio circuit card that plugs into an IBM-compatible personal computer system and software that you load from floppy disk to a subdirectory on the hard disk. This section is included to show the typical way that a third-party speech-editing system works.

The Audio Works Station product accepts audio sources from either a microphone or an audio cassette tape. It can output audio to an audio receiver and a speaker. Audio Works Station can create speech files in digital format and supports speech formats in 16-bit linear PCM, 8-bit Mu-law (64kbps PCM), and 4-bit ADPCM. It accepts only 16-bit linear PCM files for editing.

The following procedures are necessary to add speech, recorded on Audio Works Station, into the VIS.

- Extract speech from the VIS
- Edit the speech in Audio Works Station
- Transfer speech to the VIS

Each of these procedures is discussed in the sections that follow. Before performing these procedures, the list phrase command may be used to identify the characteristics of the speech file.

The following is the output from the **list phrase all in talkfile all** command which includes information on all phrases stored in the speech filesystem sorted by the respective talkfile number:

Talkfile Number	Phrase Number	Size (bytes)	Time	Coding Type	Phrase_Name
40	1085	1346	0.7	ADPCM 16	Hello
40	1086	2976	2.5	ADPCM 32	Welcome
101	1008	3451	2.8	ADPCM 32	Thank you
102	1003	2321	3.1	ADPCM 32	For using AT&T

### Retrieving Speech from the VIS

Use the following commands in the sequence shown to extract a speech phrase in the 64kbps Mu-law PCM format from the VIS system for editing on a third-party editing system. The VIS must be running in order to extract speech files from the VIS. The phrase and talkfile numbers used in the following procedure are examples used in the Audio Works Station.

1. Enter **`gse_copy 101 1008 /tmp/myfile`**

This command extracts the speech phrase 1008 in talkfile 101 to a UNIX file named `myfile` in **`/tmp`** directory. This file is converted from VIS ADPCM32 to VIS PCM64 automatically.

2. Enter **`doscpc /tmp/newfile a:/dosfile`**

This command copies the UNIX file **`/tmp/newfile`** to a file called `dosfile` in a DOS formatted floppy disk located in the floppy disk drive.

### Editing Speech in Audio Works Station

You must convert the Mu-law PCM speech stored in *dosfile* into a format supported by Audio Works Station. The VIS Mu-law PCM file format is described under the "Speech File Formats" section of Chapter 2, "Developing Speech."

After the file has been converted into an Audio Works Station compatible format, it can be edited. The modified speech needs to be stored in the VIS Mu-law PCM format in *file1* on a DOS formatted floppy-disk.

## Transferring the Speech to the VIS

To transfer the new speech phrase to the VIS system, perform the following procedure:

1. Insert the DOS floppy disk into the VIS floppy disk drive.
2. Enter **doscop a:/file1 /tmp/file1**  
This copies the speech file to file1 in the UNIX /tmp directory.
3. Enter **cat /tmp/file1 | addhdr voice > /tmp/file2**  
This adds the voice header to /tmp/file1 and stores the contents in a new file, /tmp/file2.
4. Enter **gse\_add {101 1009} adpcm32 /tmp/file2** after the voice header has been added.  
This adds the speech created on Audio Works Station to the VIS speech filing system.

---

## Recognizing Speech Input

# 4

---

### What's in This Chapter

Previous chapters have included information on planning, recording, storing, and editing speech — aspects that involve playing speech to a caller. This chapter discusses what is involved in recognizing caller input and the available speech recognition options.

As mentioned in Chapter 1, "Overview of Speech on the Intuity CONVERSANT VIS", two types of speech recognition are available. WholeWord Speech Recognition recognizes whole words, whereas FlexWord Speech Recognition recognizes subwords or parts of words. This chapter includes an in-depth discussion of both types of speech recognition.

## **Introduction to WholeWord Speech Recognition**

---

WholeWord Speech Recognition is available as an optional installable package on the Intuity CONVERSANT Voice Information System (VIS). WholeWord Speech Recognition allows you to write applications that prompt for and understand spoken input from callers. This feature supports the vocabulary “zero” through “nine,” “oh,” (synonym for zero), “yes,” and “no.”

## **Components of WholeWord Speech Recognition**

---

Components included with the WholeWord Speech Recognition package are described in the sections below.

### **International Support**

---

WholeWord Speech Recognition provides standard vocabulary support for some international languages. The international packages currently available are as follows:

- Canadian French
- Mexican Spanish

For additional information about international support, contact your AT&T representative.

### **Bilingual WholeWord Speech Recognition Capabilities**

---

You can load and operate two WholeWord Speech Recognition language packages on a single VIS simultaneously.



**NOTE:**

The bilingual speech recognition capability is available within the same call, but not within the same prompt.

### **Standard Vocabulary**

---

Table 4-1 lists the standard vocabulary for all supported languages. The WholeWord Speech Recognition feature supports spoken English from any United States region. Canadian French supports French spoken in the province of Quebec. Mexican Spanish supports spoken Spanish input collected throughout the country of Mexico.

**Table 4-1. Standard Vocabulary of Supported Languages**

<b>North American English</b>	<b>Canadian French</b>	<b>Mexican Spanish</b>
zero	zéro	cero
oh ( = zero)	—	—
one	un, une	uno
two	deux	dos
three	trois	tres
four	quatre	cuatro
five	cinq	cinco
six	six	seis
seven	sept	siete
eight	huit	ocho
nine	neuf	nueve
yes	oui	sí
no	non	no

## Standard WholeWord Speech Recognition Types

Recognition types allow you to indicate which key word or words are possible selections for each prompt. The recognition type determines which model(s) the incoming speech is compared. *Keyword* recognition types are used for isolated word recognition, that is, responses of a single word or digit.

A complete recognition type includes mathematical models of the words, a "grammar," and in some cases, a DIP. All of the recognition types and words used by an application constitute that application's "vocabulary." The standard WholeWord Speech Recognition package provides several commonly used recognition types.

The recognition type is selected in the *Mode* field on Page 2 of the Prompt & Collect screen in Script Builder as shown in Figure 4-1. The recognition types for the supported language packages are listed in Table 4-2.

*Connected-word* recognition types are used for connected-word recognition; that is, responses of more than one word or a digit string. The connected-word recognition types, with a maximum 24-digit string, are listed in Table 4-3.

---

Define Prompt and Collect		Page 2 of 3
INPUT		
Caller Input Field:	\$CI_VALUE	
No. Of Tries Used Field:	\$CI_TRIES_USED	
No. Of Digits Input Field:	\$CI_NO_DIGS_GOT	
Mode:	US_YN	TT Erase Code Active: no
Min Number Of Digits:	01	TT Erase Code Value: _____
Max Number Of Digits:	64	TT Cancel Code Active: no
TT Terminator Code Active:	no	TT Cancel Code Value: _____
TT Terminator Code Value:	"#"	No. Of Tries To Get Input: 03
TT Repeat Code Active:	no	Initial Timeout: 05
TT Repeat Code Value:	_____	Interdigit Timeout: 05

**Figure 4-1. Specifying the Recognition Type**

Table 4-2 show the Standard WholeWord Recognition Types, which are used when prompting the caller to speak a response of one word or digit. That is, callers are expected to speak only one word or one digit. The US\_1\_3 recognition type, for example, means that callers are expected to say a single word, either “one,” “two,” or “three.” A prompt that might use the 1-3 digit grammar type could be structured, respectively, as follows:

“For checking account balance, say ‘one.’ For savings account balance, say ‘two.’ For interest rates, say ‘three.’”

**Table 4-2. Standard WholeWord Recognition Types**

Digit Lengths	US English	Canadian French	Mexican Spanish
“Yes” or “No”	US_YN	CF_YN	MS_YN
One character 1-3	US_1_3	CF_1_3	MS_1_3
One character 1-3 or “no”	US_1_3N	CF_1_3N	MS_1_3N
One character 1-5	US_1_5	CF_1_5	MS_1_5
One character 1-5 or “no”	US_1_5N	CF_1_5N	MS_1_5N
One character 0-9 or “oh”	US_DIG	CF_DIG	MS_DIG

Table 4-3 shows the WholeWord Connected-Digit Recognition Types, which is used for recognizing responses of more than one digit.

**Table 4-3. WholeWord Connected-Digit Recognition Types**

String Length	US English	Canadian French	Mexican Spanish
Variable 1-24 digits of 0-9 and “oh”	US_DIG	Not Applicable	Not Applicable

The spoken string can be fixed in length from 1-10 digits, or it can be of a variable length of 1-24 digits. Grammars are provided for the 1-10 digit fixed-length and the 1-24 variable-length strings. The string length and grammar type is specified in the *Min Number of Digits* and *Max Number of Digits* fields in Page 2 of the Prompt & Collect screen shown in Figure 4-1. See *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727, in Chapter 5, “Defining the Transaction,” for more information about the **Prompt & Collect** action step.

### NOTE:

For better accuracy, the desired string length must be specified when selecting the recognition type for a fixed-length string. An area code (614), for example, consists of 3 words; therefore, the minimum and maximum values are “3.” If a variable-length string of 1 to 4 words is required, “1” is the minimum value and “4” is the maximum value.

## Models

Each word of the system’s vocabulary is represented by a mathematical *model* which contains the speech signal characteristics of the word. The model is created from thousands of samples of the spoken word. Vocabularies have one or more models for each word.

## Grammars

A *grammar*, which is built into a recognition type, is a set of rules that specifies allowable vocabulary words and vocabulary word combinations at any one point in the script (for example, “four,” “five,” “six,” “no”). After collecting spoken input, the speech recognition algorithm uses models and grammars to generate a list of candidates that most closely resemble this spoken input. The algorithm returns the most likely match to the script or, if no match fits, rejects the input. When input is rejected, the algorithm returns a “?” to the script. This return message is the same for all of the languages supported. All grammars provided in your speech recognition package share the same set of models for words that are common to their package’s vocabulary.

## **Custom Vocabulary**

---

AT&T can create a custom vocabulary to supply your application with speech recognition software for words not included in the standard package, such as "checking" and "savings."

With a custom vocabulary, the application could prompt the following:

"Please say 'checking' to hear your checking account balance, or 'savings' to hear your savings account balance."

A custom vocabulary requires application analysis by AT&T, speech data collection, model creation, and custom grammar work, and therefore, is usually expensive. However, you may want to consider FlexWord Speech Recognition as a less expensive alternative. Refer to the discussion on "FlexWord Speech Recognition Accuracy" later in this chapter.

## **Custom Recognition Types**

---

Recognition types are the choices that are associated with the Mode field on the Script Builder Prompt & Collect screen. The VIS uses the recognition type, along with other fields on the Prompt & Collect screen, to select a speech recognition grammar to be used for that prompt. Certain digit strings and custom vocabulary words may require custom recognition types for acceptable recognition accuracy. Examples include a credit card number, a merchant number, and a 10-digit telephone number. These digit strings have specific limitations on the position of certain digits within the string (for example, a telephone area code requires a number 2 through 9 for the first digit, 0 or 1 for the second digit, and 0 through 9 for the third digit). Custom recognition types help limit the recognition possibilities for the SP circuit card, which results in better accuracy. Up to 20 recognition types can fit on one SP circuit card.

## Custom DIPs

---

Enhanced speech recognition is sometimes possible through the use of a data interface process (DIP). DIPs, which are written in the C programming language, interact with your script to help access external information. Once a request is received from a TSM script, for example, the DIP processes the message and returns the results to the corresponding script. DIPs usually work based on knowledge that is unavailable to the SP circuit card.

Certain digit strings, such as credit card numbers, have check digits built into them. Verifying a check digit is done most efficiently through a DIP. DIPs may also be used in conjunction with a custom grammar, for example, to help select a variable-length digit string. Several different recognition types can use the same grammar but have different DIPs.

However, a DIP may not be necessary for many recognition types. If there is no further information to consider for a string other than what is in the grammar, a DIP is not needed. Specifically, none of the standard WholeWord recognition types, such as "yes" and "no," need a DIP.

For some recognition types, there may be trade-offs that determine how much of the structure of the input is built into the grammar and how much should be left for the DIP. For more information about WholeWord Speech Recognition types, refer to the section later in this chapter titled "Standard WholeWord Speech Recognition Types".

For more information about custom DIPs, see Chapter 4, "Data Interface Process," in *Intuity CONVERSANT VIS V5.0 Application Development*, 585-310-227.

## Word Spotting

---

WholeWord Speech Recognition also supports word spotting. Wordspotting is the ability of the VIS to search past extraneous speech during speech recognition. For example, if a caller says, "I want number five, please," the VIS recognizes the word "five" as a valid response and ignores the rest of the input. In other words, callers do not have to only speak the key word, which in this case, is "five." The caller can speak other words, and the VIS can distinguish the key word from the extraneous words.

This means that the caller's input is not limited to the words in the VIS vocabulary. For example, in the phrase "Uh, yes please," the vocabulary word "yes" is recognized if the standard vocabulary is being used by the VIS.

## **Speaker Independence**

---

The speech recognition process compares a person's voice to a set of predeveloped speech models. These models are constructed from thousands of speech samples. Regional accents and dialects associated with a particular language are incorporated into each model, as well as a mix of male and female speakers so that callers of either gender speaking in any of those dialects with any of those accents are recognized.

## **Phrase Screening (Rejection)**

---

Phrase screening or rejection is the ability of the VIS to determine whether or not a candidate key word is a close enough match to be declared a valid key word. For example, if the prompt states "Please respond with 'yes' or 'no,'" and the caller speaks the word "what," the VIS informs the caller that the response is invalid. It could then replay the prompt, "Please respond with 'yes' or 'no.'"

## **Recognition Confirmation**

---

Recognition confirmation is the ability of the VIS to repeat the caller's spoken response back to the customer for confirmation. For example, a script prompts a caller "Please say your area code," and the caller says, "Six, one, four." Using recognition confirmation, the VIS repeats what it recognized the caller's response to be and then requests, "If this is correct, say 'yes,' if not say 'no.'" The caller then confirms whether the system recognized the spoken input properly.

## **Barge-In**

---

Barge-in, also referred to as Recognize During Prompt, is the ability of the VIS to allow callers to interrupt or barge in during voice playback by speaking a vocabulary word. Speech recognition accepts either speech or touchtone input in response to a prompt. Barge-in operates much like the talkoff option for touch-tone input, where a caller can interrupt the VIS by pressing a touch tone. Experienced callers appreciate being able to shorten the transaction time by not being required to listen completely to each prompt. Barge-in can be enabled or disabled at each prompt in your application as determined by you. For more information about the talkoff option for touchtone input, see the *Intuity CONVERSANT VIS V5.0 Application Design Handbook*, 585-310-551.

### **⇒ NOTE:**

The VIS is able to detect touch tones immediately; however, it does take the VIS a few seconds longer to detect good speech, as opposed to a cough, sneeze, etc. You should expect a slight delay when using the barge-in feature.

For isolated word recognition, the prompt will not stop until the VIS recognizes a valid vocabulary word. Once the prompt completes playback, the Initial time-out field eventually ends the recognition if no valid input is received.

For packages that support connected-digit recognition, the prompt playback stops between the recognition of the first and last word of the input, at a point determined by the ability of the VIS to decide when valid input has started.

You can enable or disable the barge-in feature within the Prompt & Collect action step. When enabling, any prompt with the Recognize During Prompt field set to “yes” on page 1 of the Prompt and Collect screen is interrupted by voice or by touch tones. (See Chapter 6, “Using Optional Features with Script Builder,” of *Intuity CONVERSANT VIS V5.0 Script Builder*, 585-310-727. If your script does not use *Recognize During Prompt*, be sure your prompts are worded carefully to make sure that callers know they cannot speak until the prompt is finished. For instance, look at the wording and the <pause> length of the following prompts. There are no large gaps of silence to tempt callers to speak during the menu prompts. These prompts are designed for use when barge-in is disabled so that callers cannot barge-in and must wait until the prompt is finished.

#### **Example 1 — Discouraging Barge-In**

“For sales say, ‘one.’ <a short pause>  
For service, say, ‘two.’ <a short pause>  
To speak with a representative, say ‘three.’ ”

#### **Example 2 — Discouraging Barge-In**

“You may order up to five copies. Please say how many copies you would like.”

#### **Example 3 — Encouraging Barge-In**

These next prompts are designed to encourage callers to barge-in when barge-in is enabled. Note that the pauses are longer, so callers are encouraged to respond after hearing the desired option.

“For sales say, ‘one’ now. <a 1-1.5 second pause>  
For service, say, ‘two’ now. <a 1-1.5 second pause>  
For a representative, say ‘three’ now. <a 1-1.5 second pause>

#### **⇒ NOTE:**

Please note that if there is silence at the end of a prompt, the caller may speak but the system may not be prepared to listen. Make sure that there is no silence at the end of your recorded phrases when barge-in is disabled.

---

## **Dual Tone Multifrequency (DTMF) Support**

---

Even with WholeWord Speech Recognition installed, callers still have the option of responding to prompts with touch tones rather than speaking.

## **Connected Digit Recognition**

---

Applications can accept spoken digits in two ways. “Connected digits” are strings of digits, spoken one after the other with no long pause in between. For packages which support connected-digit recognition, you can collect input by specifying a variable or fixed-length string within a determined range of up to 24 connected digits. Recognition is better for fixed length strings than for variable strings.

Tone-paced digits are strings of digits spoken one at a time. The caller speaks a digit then waits for a tone before saying the next digit.

### **⇒ NOTE:**

Refer to Table 4-3 for the supported connected-digit recognition packages. Note that connected-digit recognition is only supported for U.S. English. Packages without connected digit recognition should use tone-paced recognition (that is, Canadian French and Mexican Spanish).

Number entry is an essential component if you want your callers to input numbers greater than one digit in length, (for example, “1-3-5” or “2-4-6-8”). But the spoken input has to be collected singularly if your package does not support connected digit recognition. (See the note that follows this section, if your package supports connected digit recognition.) If your package does not support connected digits, then you must piece together caller tone-paced input in order to create a digit string. A four-digit number, for example, could be collected as follows:

- “Please say the 4-digit extension of the person you wish to call. After you say each digit, wait for a tone before saying the next.”

Callers inputting the three-digit string, 6-1-4, might say the following:

- (beep) “six” (beep) “one” (beep) “four”

### **Fixed-Length versus Variable-Length Connected Digits**

The capability to recognize fixed-length strings of one to ten digits is provided with WholeWord Speech Recognition packages that support connected digit recognition (that is, U.S. English; See Table 4-3). A fixed length string is a string of digits that is always made up of the same number of digits. For example, a social security number is a fixed-length number, since it always consists of nine digits. You can specify a string length to be fixed or variable within a range. A variable length string indicates a string of digits that varies. For example, a house number in a street address is a variable-length string.

The accuracy of this WholeWord feature decreases with strings longer than 10 digits. Accuracy is best when you use fixed-length strings. Whenever possible, try to specify the length of the string to increase recognition accuracy.

If your application must accept a variable-length digit string, you may be able to increase recognition accuracy by using a two-step entry process. First ask for the number of digits, then ask for the digits. The recognizer then knows how many digits to look for. The recognizer is the process within the VIS that compares caller input to internal models and returns a match to the caller. For example,

“How many digits are in the next code?”

<3>

“Please say the three-digit code now.”

---

## **WholeWord Speech Recognition Accuracy**

---

WholeWord Speech Recognition accuracy depends not only on the recognition algorithms, but also on the models, grammars, DIPs, prompt structure, calling population, and the recognized data itself. Each of these factors can impact recognition accuracy positively or negatively. Also, accuracy must be based across the entire calling population. Therefore, any attempt to measure accuracy must include a statistically representative sample of the calling population.

### **Positive Influences on Wholeword Speech Recognition Accuracy**

---

The following items have a positive impact on WholeWord Speech Recognition accuracy:

- Isolated word recognition  
Isolated word recognition is very high. In addition, the smaller the number of choices in an isolated word recognition type, the better the accuracy. For example, US\_1\_3 is more accurate than US\_1\_5, which in turn is more accurate than using US\_DIG with length 1.
- Fixed length digit string  
For connected digit recognition, a fixed length recognition type provides better accuracy than a variable length recognition type. If possible, variable length strings should be avoided in WholeWord Speech Recognition applications.
- Validation of data  
Try to verify the recognized result against a database or a host field. This will help improve the overall accuracy of your application, especially when a longer string is input.
- Reprompt  
If the keyword is not spoken, and the VIS does not misinterpret extraneous words for the keyword, the system can reprompt the caller. If the accuracy measurement is based on a WholeWord application with confirmation and reprompt step, the accuracy will increase.

- Prompt structure

The prompt structure can greatly affect accuracy by promoting a calm, clear voice, helping the caller barge-in at the appropriate time or wait until the prompt is complete before talking (when barge-in is disabled), and providing consistent instructions on what the caller should say to get the desired result.

- Menu prompts

For best results, menu prompts should be built with the following structure:

<desired result> <action required>

For example:

“To hear your checking account balance, say 1.”

“To hear your savings account balance, say 2.”

By speaking the action required at the end of the prompt, the caller will not have to remember what is required of them through the description of the desired result. This format, <desired result> <action required>, is accepted better by callers than prompts structured in the reverse order (for example, “Say 1 to hear your checking account balance”). In addition, if you wish to encourage your callers to barge-in when they hear their desired result, you can add a small phrase after the action required phrase.

- Yes/No prompts

A yes/no prompt should be structured as a yes/no question; that is, in the same way that you would ask a question. For example:

“Would you like to hear your order again?”

If the caller does not respond to the prompt, the follow up prompt could be as follows:

“Would you like to hear your order again? Please say ‘yes’ or ‘no’.”

This wording is more natural than the following:

“If you would like to hear your order again, say ‘yes.’  
Otherwise, say ‘no’.”

Once again, if you wish to encourage the use of barge-in, add a small pause (about 2.5 seconds) following the action required phrase. For example:

“Would you like to hear your order again? <pause>  
Please say ‘yes’ or ‘no’.”

- Experienced calling population

In an application where the calling population is closed and the callers become experienced or have been trained, recognition accuracy will improve.

- Custom grammars and DIPS

Custom grammars improve the Signal Processor's (SP) ability to "score" the candidate, by selectively limiting the recognition possibilities. Scoring the candidate is the SP's process of assigning a probability rating to the recognized input and sending the result that scored that highest back to the script. Custom DIPS help further process the recognition result with information unavailable to the SP circuit card. However, not all applications can take advantage of custom work.

- Informative prompts

Lengthy prompts that provide detailed instructions on how to respond may improve accuracy, but are generally unacceptable from an application perspective, unless the application has infrequent users. Users who only use interact with VIS prompts infrequently (for example, once or twice a year) are more willing to listen to a lengthy prompt.

## **Negative Influences on Wholeword Speech Recognition Accuracy**

---

The following items have a negative impact on WholeWord Speech Recognition accuracy:

- **Word spotting with variable length strings**

Word spotting provides the ability to filter out non vocabulary words from a caller's response for fixed-length recognition types, but will not provide accurate filtering when using variable length recognition types. Accuracy will not be acceptable if callers speak extraneous, non-vocabulary words while the system is trying to recognize a variable-length string.

- **Information type**

Recognition of data that is normally not spoken in digits 0-9 will adversely affect accuracy. For example, dollar amounts and days of the month are usually not spoken in digit form 0-9. To speak the date December 15, the caller would be required to say "1-2-1-5." Training callers to speak information in this format may increase recognition accuracy. However, if callers speak natural numbers, such as "fifteen," speech recognition will not work.

- **Strong regional or national accents or dialects**

Although WholeWord Speech Recognition is based on thousands of speech samples per word, strong accents or dialects can be misinterpreted by the VIS in some instances.

- **Connected digit-string length**

Recognition works best for short digit strings (less than five digits). With each additional number in a string, the overall accuracy falls slightly. To improve accuracy for longer digit strings, the string could be broken up into logical units, such as the area code and seven-digit phone number. Accuracy deficiencies must be taken into consideration when planning digit string Speech Recognition of more than 10 digits.

The following list shows the correlation between recognition accuracy and connected digit string length. A 3% accuracy degradation occurs for each additional digit required to be interpreted. Further extrapolation can reveal accuracies for more than 5 digits.

- 1 digit = 97%
- 2 digits = 94%
- 3 digits = 91%
- 4 digits = 88%
- 5 digits = 85%

With any language, this function is application dependent, although recognition accuracy improves significantly if a fixed length string less than 10 digits in length is used.

- Environment

A very noisy environment, such as an airport or train station, can cause recognition accuracy problems. In certain cases, speech data can be collected to build custom word models based on the noisy environment to improve recognition accuracy.

- Extraneous words within response

Extra words spoken alongside the keyword can sometimes lead to misinterpretation by the VIS if they have the same characteristics as the keyword.

It is important to understand that there are other limitations to the performance of the WholeWord Speech Recognition feature. If the VIS is under-engineered for a particular application, it may not perform to your satisfaction. The capability of the VIS and WholeWord Speech Recognition is application dependent. Several factors can affect the number of channels available for speech recognition. These factors can be related to a particular application. There are also system factors that relate to all applications.

Application-related factors that affect the number of supported WholeWord Speech Recognition channels include:

- The percentage of time spent recognizing speech input, for example, how much speech data must be gathered during the application script.
- The percentage of callers who use touch-tone entries require far less hardware and software resources to process.
- The number of simultaneous speech recognition calls expected
- Using the barge in or *Recognize During Prompt* option with WholeWord Speech Recognition increases the amount of hardware and software resources to process each transaction.
- Using custom recognition types, a DIP, or vocabularies

System-related factors that affect the number of supported WholeWord Speech Recognition channels include:

- Using speech coding (voice capture)
- Using call event tracking excessively (more than 40 events stored)
- Reading large (more than 50 records) database tables that are not indexed
- Excessive reading and writing to database tables (see the "Database Environment" chapter of the *Intuity CONVERSANT VIS V5.0 Operations*, 585-310-550)
- Sending and receiving many host screens
- Expected simultaneous call volume
- Requesting ORACLE reports during peak call periods

---

## How WholeWord Speech Recognition Works

---

This section describes the processing involved during WholeWord Speech Recognition on the Intuity CONVERSANT VIS.

### Speech Input from Caller

A person calls the VIS and the VIS answers the phone. At the beginning of the script, the application developer usually places an **SR\_Prompt** instruction to enable barge-in. If a barge-in resource is available, it is reserved for this call. Otherwise the return code to **SR\_Prompt** is negative, and barge-in will not be available. During a **Prompt & Collect** action, the system prompts the caller for a response. If the **Prompt & Collect** action allows for voice input, the VIS locates and reserves a free Signal Processing (SP) circuit card resource. If the caller responds using touch tones, no SP resources are required for this particular prompt and the resource is freed. If the caller responds using voice input, the input is sampled at 8000 samples/sec, digitized, and then transferred to the Signal Processing circuit card via a Time Division Multiplexed (TDM) Bus. The TDM bus provides a communication link between the SP and the telephone interface cards (T1 or Tip/Ring) and also allows the SP resources to be shared across all incoming voice channels. This helps provide more channels of recognition, since there is no need to dedicate certain incoming channels to only WholeWord Speech Recognition when accepting both touch tone and voice input.

### The Recognizer

The SP and CMP circuit cards are loaded with the WholeWord Speech Recognition software, which is often referred to as the "recognizer." The recognizer compares the incoming speech sample to only the word models indicated by the recognition type chosen in the **Prompt & Collect** action. The recognition type is found in the Mode field on the second page of the **Prompt & Collect** statement. The recognizer computes a score for each of the models. The score represents the probability that the incoming speech matched a word model indicated by the recognition type. The recognizer returns the most likely candidate to the recognition dip, and the dip returns the result to the script as \$CI\_VALUE. If no candidates sufficiently match, the input is rejected and a "?" value is returned to the script. For example, for a prompt of "Please say yes or no," the application developer would select the US\_YN recognition type, which is used for "yes" or "no" responses. The caller's voice response to the prompt would be compared to both a model of the word "yes" and a model of the word "no." A score would be computed for "yes" and for "no." Whichever score was highest would determine the recognized result. A value of "Y" or "N" would be returned in \$CI\_VALUE. If neither scored high enough, the recognizer would return a "?" in \$CI\_VALUE. The application would then decide what to do next, based on the value of \$CI\_VALUE. For strings of digits, the recognizer first determines the top

four scoring strings, and their resulting scores. The recognition dip then returns the top scoring string whose length falls between *Min Number of Digits* and *Max Number of Digits* to the script, which is stored in \$CI\_VALUE.

### Computational Processing

AT&T provides a Companion (CMP) circuit card to help the SP board with the intensive processing required for recognition algorithms. The CMP circuit card has an additional 12 digital Signal Processors and is connected to the SP circuit card via a bus. One SP with one CMP circuit card can provide up to 6 shared channels of simultaneous connected digit recognition. With the maximum configuration of two CMP circuit cards, the SP can provide up to 12 shared channels of simultaneous connected digit recognition.

### Further Processing

There is a Data Interface Process (DIP) associated with Whole Word Speech Recognition, called `/vs/bin/vrs/recog_dip` which is also known as DIP25, or the **recog\_dip**. The **recog\_dip** is actually an optional, post-processing mechanism for enhancing the performance of some recognition tasks. It is intended to furnish additional recognition processing beyond what is already provided by the models and grammars of the recognizer. Whether or not the **recog\_dip** is used during a particular recognition task is determined in the `/vs/data/sr_file` file, to be discussed in the next section. When the **recog\_dip** is in use, the recognizer sends its top 4 choices (and a score for each) to the DIP, which in turn makes a call to the `/att/asr/find_best.c` routine. The **find\_best.c** routine is responsible for making a final selection from the list of recognition choices and passing it back to the script.

The default version of the **find\_best.c** routine (the one that comes with the recognition package) simply selects the result with the highest score that falls within the Min and Max digit constraint. Therefore, if a fixed length grammar is being used by the recognizer, the default version of the **find\_best.c** routine adds no value to the recognition procedure. On the other hand, if recognizer is applying a variable length grammar, then the default **find\_best.c** is useful for range checking.

The real power of the **recog\_dip** becomes apparent when the **find\_best.c** routine is customized to place added constraints on the recognizer. This helps limit the search of possible recognition results and thereby improves recognition accuracy. For example, if an application is attempting to recognize a ten digit phone number and it knows that when the area code is 614, the final four digits of the phone number will always fall between 5500 and 9000, and when the area code is 216 or 513 the final four digits will always be between 1000 and 4400, the **find\_best.c** routine can be customized to account for these data dependencies, which should be translated to an increase in recognition performance. The point to note here is that a certain amount of dependency between digits can be accomplished with a custom grammar, which gets implemented at the recognizer level. However, more complex patterns or dependencies within the expected user input is best handled with the **find\_best.c** routine and the **recog\_dip**.

After making and compiling changes made to **find\_best.c**, you will need to remake the **recog\_dip**. Because of this, a makefile is included in the **/vs/bin/vrs** directory. Any changes made should be tested thoroughly, as mistakes could severely impact your application's recognition accuracy.

## Introduction to FlexWord Speech Recognition

---

The FlexWord Speech Recognition package can be used to recognize callers speaking words from a specific vocabulary defined by you. A word is any word or phrase that can be recognized with FlexWord Speech Recognition and must be associated with a wordlist. A wordlist is a set of words that can be recognized with FlexWord Speech Recognition by any **Prompt & Collect** action step. A vocabulary is a set of wordlists associated with a particular FlexWord application script.

FlexWord Speech Recognition relies on sub-word technology. Sub-word technology relies on phonemic recognition for analyzing and recognizing American English vocabularies.

### Defining Phonemes

---

Phonemes are blocks of sound which form recognizable words when strung together in a particular order. The English language contains 40 phonemes that represent all basic sounds used with the language. The word "one," for example, consists of three phonemes: "w-uh-n." For more information about phonemes, refer to "Using the FlexWord Toolkit to Create Your Wordlists" later in this chapter.

As introduced in Chapter 1, "Overview of Speech on the Intuity CONVERSANT VIS", phonemes are blocks of sound that form recognizable words when strung together in particular orders. FlexWord Speech Recognition uses the 40 phonemes illustrated in Figure 4-2.

---

E heed	A hay	p pick	T thin
i hid	I hide	t tick	D then
e head	O hoe	k kick	s sip
a had	Y boy	b bit	S ship
o cot	R her	d dot	C chip
u hook	W how	g got	J jip
U rue	r rip	m met	z zip
> saw	l lip	n net	Z measure
^ cup	w we	N ring	f fit
& data	y yes	v vet	h hit

---

**Figure 4-2. Illustration of the 40 American English Phonemes**

The FlexWord recognizer looks for phonemes when recognizing a word or phrase. The maximum number of words or phrases that can be loaded onto the FlexWord recognizer is 2000. FlexWord supports up to 200 wordlists and 500 words per wordlist as shown in Table 4-4. However, every word or phrase in a wordlist is counted as a distinct and separate word. If the word "help" appears in 10 of your wordlists, "help" will be counted as 10 separate words. However, phrases, such as "Loan Information," count only as one word. Each wordlist must be given a name tag consisting of 1-14 upper-case characters. For more information about word, wordlist, and vocabulary, refer to "Understanding the Speech Filesystem" in Chapter 1, "Overview of Speech on the Intuity CONVERSANT VIS".

**Table 4-4. FlexWord's Maximum Capacity**

Words/phrases	= 2000
Wordlists	= 200
Words per wordlist	= 500
Characters per wordlist name tag	= 14

## **Components of FlexWord Speech Recognition**

---

Components of FlexWord Speech Recognition are discussed in the following sections.

### **AT&T Custom Vocabulary Service**

---

The AT&T Custom Vocabulary Service is available to build your FlexWord Speech Recognition vocabularies for a fee.

### **FlexWord Toolkit**

---

FlexWord supports fast, low cost delivery of speech recognition vocabularies on a custom basis through the use of the FlexWord Toolkit, an optional package that supports unique or volatile applications such as name dialing, menu selection by words, command words, and entry of client and department names. This optional toolkit gives you a point and click, graphical environment for adding, deleting, or changing words on new or existing wordlists. A benefit of using the FlexWord Toolkit is that you can use these wordlists in your application immediately after you create them.

### **Extended Vocabularies**

---

A FlexWord vocabulary can include up to 200 wordlists, and one wordlist can include up to 500 words. A word may be used in more than one word list, but each use counts against a limit of 2000 words per system. Each prompt in a FlexWord script requires a wordlist, which is a list of valid words that the caller could respond with. For example, if the script played a prompt that said "Would you like information on your checking account or savings account?," a wordlist including the entries "checking" and "savings" would be needed.

The FlexWord Speech Recognition package supports a total vocabulary of 2000 words. For example, the following are a few of the possibilities:

- 4 wordlists with 500 words each for a total of 2000 words
- 50 wordlists with 40 words each for a total of 2000 words

If you have an application that requires more than 2000 words, contact your AT&T representative for assistance.

## **Keyword Spotting**

---

The FlexWord Speech Recognition standard vocabulary package can recognize or spot a key word within a spoken phrase. The system filters out extraneous speech or noises (such as a cough) during recognition. This means that the caller's input is not limited to the words in the VIS vocabulary. For example, if one of the words in your wordlist is "checking," then the VIS can recognize it when a caller says "checking account please."

## **Phrase Screening (Rejection)**

---

The FlexWord Speech Recognition package can be programmed to alert and reprompt the caller if the spoken response was not recognized by the VIS. For example, if the prompt stated "Please respond with 'checking' or 'savings,'" and the caller spoke the word 'banking,' the VIS could inform the caller that their response was invalid and would replay the prompt "Please respond with 'checking' or 'savings.'" "

## **Limitations of FlexWord Speech Recognition**

---

The FlexWord Speech Recognition package includes the following limitations.

### **Barge-In Not Supported**

---

The ability of a caller to immediately speak their response before the menu prompts are finished is not allowed. A caller must wait until the system is completely finished prompting for information before each entry is spoken.

### **Lack of International Support**

---

International languages are not supported through FlexWord Speech Recognition. The only language supported is American English.

Refer to Chapter 5, "Putting It Together", for information about getting the most out of your FlexWord Speech Recognition application.

## **Preparing a FlexWord Application**

---

This section includes information about preparing a FlexWord Speech Recognition application, as well as information about how you can create the vocabulary to be used with your application.

### **Designing the FlexWord Application**

---

To begin with, you need to draft a design of your desired FlexWord application. When responding to a prompt, the caller answers with specific words. You must define the words you wish the system to recognize and group these words into “wordlists.” Each prompt should have its own wordlist. In Figure 4-3, each menu prompt has a corresponding wordlist. The first menu prompt looks to the “Information” wordlist to verify the caller’s first request. The second menu prompt looks to the “Loan” wordlist to verify the caller’s second request. The wordlists in both Figure 4-3 and Figure 4-4 contain groups of possible words from which callers can choose.

### **Using Wordlists in Scripts**

Each application may use several wordlists. On the second screen of your Prompt & Collect action step, you must specify the name of the wordlist to be used for the prompt in the *Mode* field. You see all available wordlists from the Prompt and Collect screen. FlexWord looks to the wordlist that you have designated by name in the *Mode* field of the Prompt and Collect screen.

The directory **/att/asr/wordlists** is the directory for all wordlists, whether you create them yourself with the FlexWord Toolkit or AT&T creates them for you. Wordlist files are named according to wordlist name tags; thus, a wordlist with the name tag “Loans” is listed as the file *LOANS* in the *Mode* field of the Prompt and Collect screen. The actual content of each wordlist file includes the English spelling and the phonetic breakdown of all of the words on the specific wordlist. The words “Account Information,” “Mortgage Information,” and “Loan Information,” for example, would be included in the Loans wordlist file in the wordlist directory.

FlexWord scans for specified words; it supports word spotting, that is, it has the ability to search past extraneous speech during a recognition. For example, if you give a caller the menu choice “account” and he or she specifies “account information,” FlexWord would recognize and accept the word “account.” For this example, it is not necessary to have both “account information” and “account” on the wordlist.

After receiving spoken input, the FlexWord recognition algorithm scans the appropriate wordlist and generates a group of candidates that most closely resemble the caller’s response. The algorithm returns the most likely match to the script. Because FlexWord has phrase screening (rejection), FlexWord can reject unrecognized speech. Defining your wordlist is very important for application success.

It is also possible to consolidate wordlists. For example, if the first Prompt and Collect screen allows for colors “red” and “blue” and the second Prompt and Collect screen allows for choices “green” and “purple”. You can put “red, blue, green, purple” on the same wordlist and direct the application to look at the wordlist with the name tag “Colors” for both prompts. Then, if the caller chooses “green” at the first prompt (which only allows for choices “red” and “blue”), the script could evaluate “green” as an unacceptable choice and you can reprompt the caller.

 **NOTE:**

All of the wordlists for all of your applications constitute the “vocabulary.”

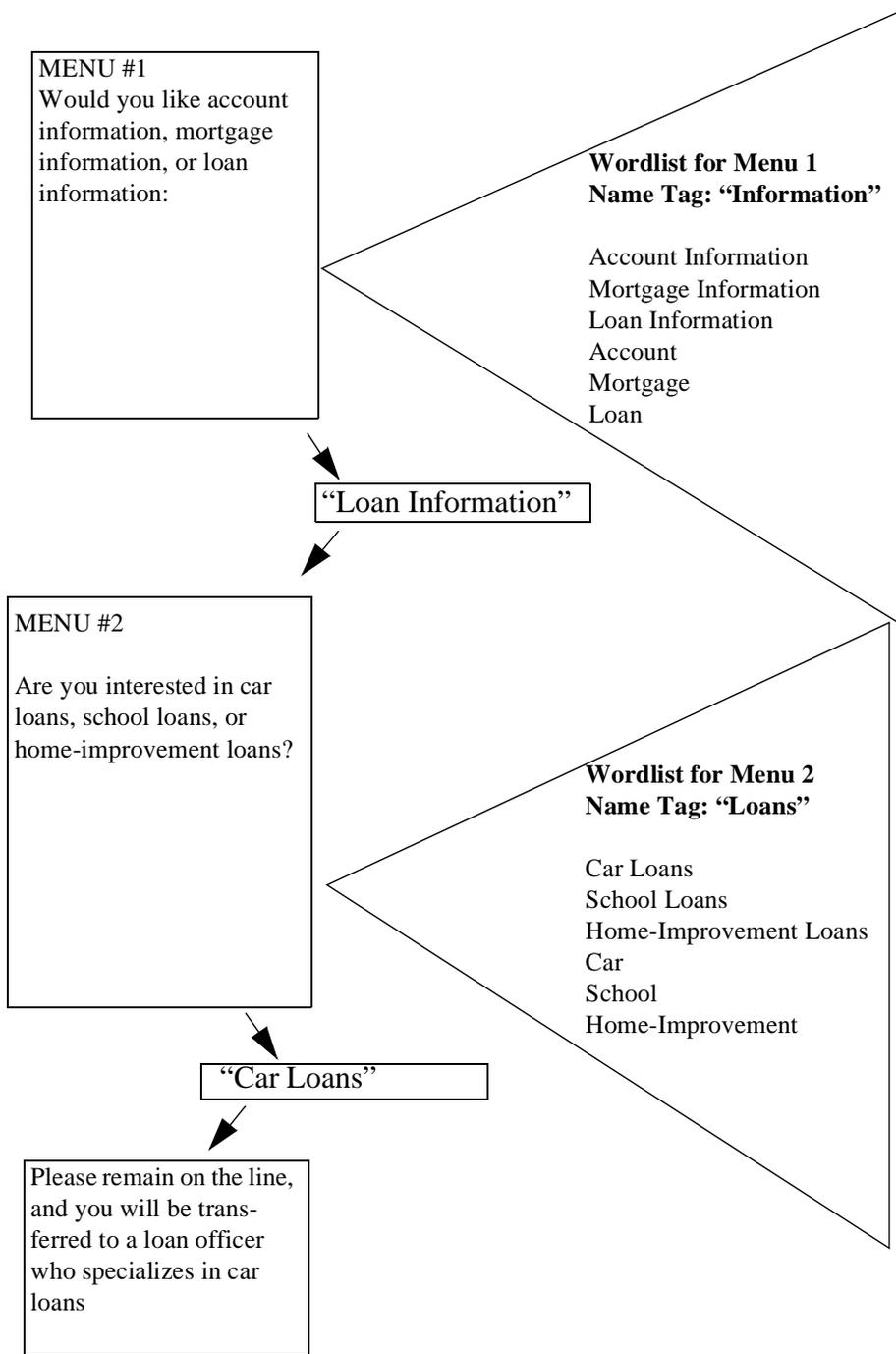


Figure 4-3. FlexWord Application Example with Menu Prompts

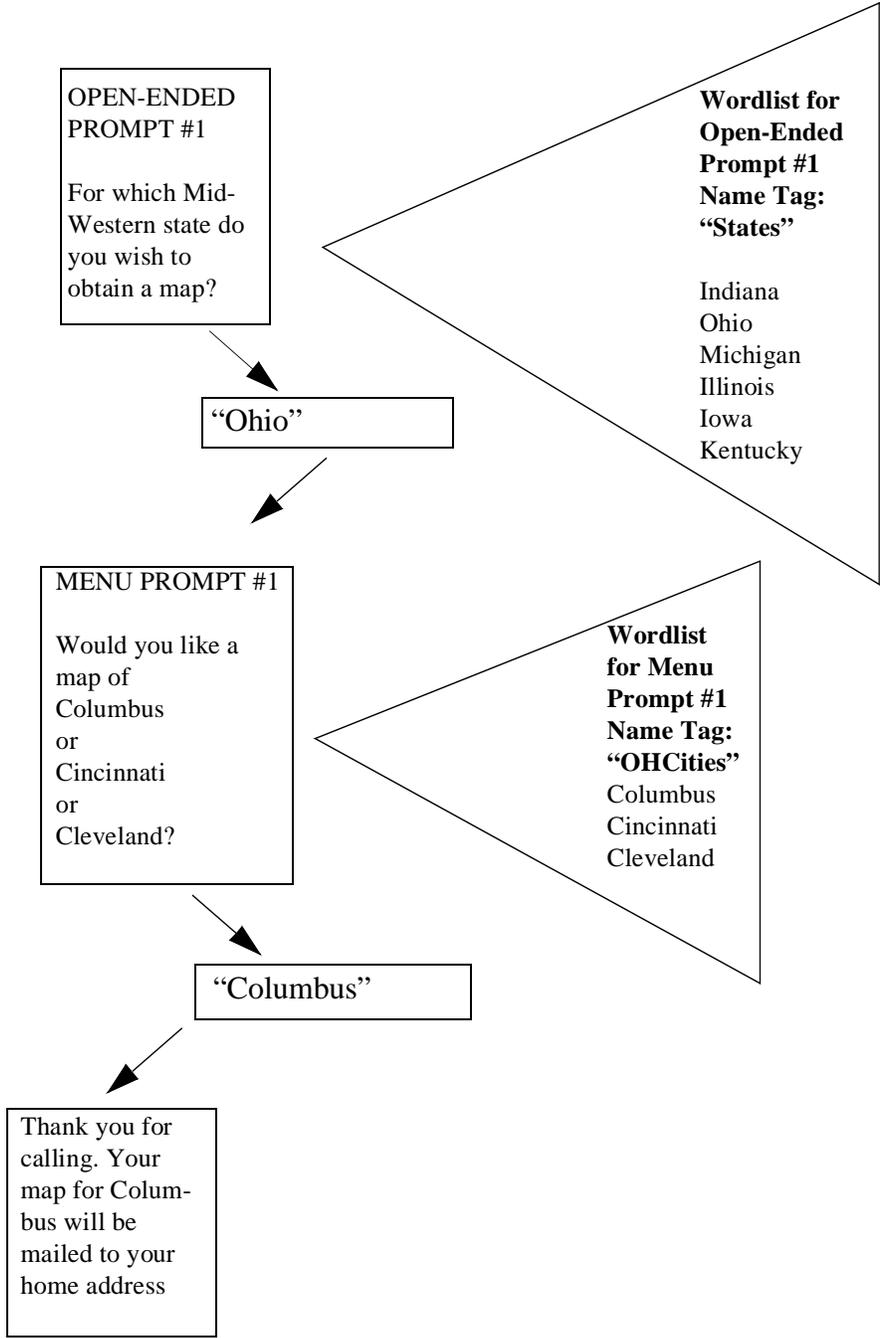


Figure 4-4. FlexWord Application Example Using Open-Ended and Menu Prompts

## Creating Wordlists

---

This section includes information about the two ways in which you can create FlexWord Speech Recognition wordlists.

### Purchasing an AT&T Custom Speech Package

This procedure involves preparing a FlexWord application and then contacting AT&T for the customization.

- Contact AT&T's FlexWord technical support staff on (404) 242-1551 after deciding upon and naming your wordlists. Inform us of your wordlists and vocabulary, and we will work with you to customize your vocabulary package. We will also answer any questions you might have.
- When the customizing process is complete, AT&T sends you your FlexWord package with your customized wordlists.
- Once you receive your customized wordlists, you can begin implementing your application.

### Using the FlexWord Toolkit to Create Your Wordlists

The FlexWord Toolkit allows you to construct words and phrases by stringing phonemes together. A standard dictionary is included with the Toolkit as well as three commands for creating and modifying wordlists. The sections that follow include information about hardware and software requirements, configuring the hardware, and creating, editing, and deleting wordlists.

**⇒ NOTE:**

You cannot use the FlexWord Toolkit with releases prior to Intuity CONVERSANT VIS Version 5.0. Also, you cannot build wordlists with the FlexWord Toolkit and use them on an Intuity CONVERSANT VIS Version 4.0 system.

### **Hardware and Software Requirements**

The software requirements for the FlexWord Toolkit are as follows:

- FlexWord Toolkit package
- Text-to-Speech (TTS) package
- Mouse driver software

Refer to *Intuity CONVERSANT VIS V5.0 Software Installation*, 585-310-151, for more information.

The hardware requirements are as follows:

- Tip/Ring (T/R) circuit card (AYC5B, AYC6B, or AYC28)
- Speaker/amp system or phone line to the T/R circuit card
- SP circuit card (AYC9) with TTS functionality assigned
- Mouse

Refer to *Intuity CONVERSANT VIS V5.0 MAP/100 Hardware Installation*, 585-310-148, *Intuity CONVERSANT VIS V5.0 MAP/100C Hardware Installation*, 585-310-149, or *Intuity CONVERSANT VIS V5.0 MAP/40 Hardware Installation*, 585-310-150, for more information.

### **Configuring Hardware for the FlexWord Toolkit**

This section includes configuration information for the FlexWord Toolkit.

- **Configuring the T/R circuit card**

The T/R circuit card plays the phonetic pronunciation of the words in a wordlist. The T/R circuit card must be either a AYC5B, AYC6B, or AYC28. It is most convenient to configure your T/R circuit card with OS Index 0, however, it is not required. If you don't configure your T/R circuit card with OS Index 0, you just have to specify the correct channel number. For more information about the T/R circuit card, refer to the Intuity CONVERSANT VIS Version 5.0 Hardware Installation book specific for your platform.

- **Configuring the speaker/amp system**

If you use an external speaker/amp system for phonetic pronunciation from within the FlexWord Editor, the AUDIO OUT jack on the T/R circuit card should be connected to the speaker/amp. The AUDIO OUT jack is located on the faceplate of the T/R circuit card. For more information about the T/R circuit card, refer to the Intuity CONVERSANT VIS Version 5.0 Hardware Installation book specific for your platform.

- **Connecting the Phone Line**

If you use a phone line for the phonetic playback within the FlexWord Editor, the default for the phone line is to be connected to Port 1 on the T/R circuit card (that is, the lowest channel T/R circuit card 0). The port is the first modular jack labeled 1,2,3 and provides access to channels 0, 1, and 2. Refer to the Intuity CONVERSANT Hardware Installation book specific for your platform for an illustration of Port 1 on the T/R circuit card.

The AUDIO OUT jack on the T/R circuit card should be physically disconnected from any speaker/amp system.

To use a phone connection with the FlexWord Editor, simply place a call to the phone number associated with channel 0 after you initialize the editor. The FlexWord Editor will take the line off-hook and can then communicate to the user through the phone handset.



**NOTE:**

Refer to Appendix C, "Computing Channel Numbers", for the procedure on how to compute channel numbers if you are not using the default.

- **Configuring the Text-to-Speech Circuit Card**

Once configured for TTS, the SP circuit card generates the phonetic pronunciation from within the FlexWord Editor. Configuring your SP circuit card for TTS functionality involves several steps. First, you must determine which SP circuit card should be assigned Text-to-Speech functionality. The SP circuit card must be an AYC9. To determine which AYC9 SP circuit card should be assigned TTS functionality, perform the following procedure:

1. Enter the following command: **display card sp**

An example of the output appears as shown below:

---

```

Card 2 is SP-8   O.S.Index: 0       Function: VOICE
                  State: Inseru    Options: slave,tdm1

Card 3 is SP-2   O.S.Index: 1       Function: VOICE
                  State: Manoos     Options: slave,tdm1
CMP 0           State: Not_diag
    
```

---

**Figure 4-5. Output from Display Card Command Prior to Assigning TTS Functionality**

The AYC9 SP circuit card that should be used for TTS functionality is designated as SP-8, which is an 8Mb-SP circuit card. Notice that SP-8 is assigned to Card 2 in Figure 4-5.

You will also need to set your TTS-SP circuit card to the correct OS Index. The O.S. Index is the address used by the VIS for accessing a particular circuit card. Notice that the O.S. Index is set to 0 in Figure 4-5.

2. Interpret the results of the **display card** command. Make a note of the card number and the O.S. Index.

---

```
Card 2 is SP-8   O.S.Index: 0   Function: TTS
                State: Inseru   Options: slave,tdm1

Card 3 is SP-2   O.S.Index: 1   Function: VOICE
                State: Manoos   Options: slave,tdm1
    CMP 0        State: Not_diag
```

---

**Figure 4-6. Output from Display Card Command After Assigning TTS Functionality**

**⇒ NOTE:**

Before you begin using the FlexWord Toolkit, verify that all circuit cards are inservice. For more information about placing circuit cards inservice, see the *Intuity CONVERSANT VIS V5.0 Operations*, 585-310-550.

- **Installing and Administering the Mouse**

For instructions on installing the mouse, refer to the *Intuity CONVERSANT VIS Version 5.0 Hardware Installation* book specific for your platform. For instructions on administering the mouse, refer to *UnixWare System Administration System Setup and Configuration*, 585-350-908.

**⚠ CAUTION:**

When referring to Chapter 1, “Setting Up the Work Environment,” in *UnixWare System Administration System Setup and Configuration*, 585-350-908, in the section “Adding a Mouse Device,” note the following:

In order to save the changes to the mouse configuration that you have just added, you need to select **U** (Update Mouse Configuration and Quit) and then enter **mouseadmin** before you can test the mouse configuration by selecting **T** (Test your mouse configuration). The procedure is shown in the reverse order in Chapter 1, “Setting Up the Work Environment.”

## Creating and Modifying FlexWord Wordlists

Before using the FlexWord Editor, make sure you have configured all of the necessary hardware discussed in the previous section, "Configuring Hardware for the FlexWord Toolkit".

This section includes how to create vocabularies and wordlists, add words and phrases, delete words and phrases, and make phonetic changes in a wordlist by using the following commands:

- **wl\_init**

The **wl\_init** command creates the phonetic spelling for each word or phrase in a wordlist.

- **wl\_edit**

The **wl\_edit** command invokes the FlexWord Editor, which is a phoneme editor that allows you to hear and change the pronunciation for each of the words in your file.

- **wl\_copy**

The **wl\_copy** command copies wordlists to disk.

### NOTE:

The VIS must be stopped in order to use the FlexWord Toolkit. If the VIS is running, the Toolkit will ask you if you would like to stop it. Only a "yes" will permit you to run the FlexWord Editor.

To create a FlexWord wordlist, perform the following procedure:

1. Select a working directory.

You may want to create a directory with a name similar to the wordlist that you want to create.

2. Create a file in the working directory using vi or some other editor that contains all of the words you would like to recognize at a particular prompt. The following rules apply when using the FlexWord Editor:

- The filename of the wordlist must be upper case, alphanumeric, and 14 characters or less in length.
- Each word or phrase in the wordlist must be on a separate line.
- White space is not allowed, and you must separate words in a phrase with an underscore (\_).
- Legal characters are a-z, A-Z, underscores("\_"), and apostrophes ("")
- Illegal characters are !, (, ), ?, and .
- Refer to Figure 4-4 for the maximum number of words in a wordlist

If you are creating wordlists for the first time, refer to the section earlier in this chapter, "Using Wordlists in Scripts" for tips on creating wordlists.

### 3. Enter `wl_init <filename>`

The `wl_init` command adds a phonetic pronunciation to each of the words and phrases in the file.

System response:

File *filename* now contains initial phonetic breakdown  
Enter `cat <filename>` to view the phonetic spelling of each word/phrase in your file. Refer to Figure 4-7 for an example of the output.

#### NOTE:

You can also view the phonetic spelling of a word once you invoke the FlexWord Editor by clicking on a word in your wordlist and looking at the phonetic spelling in the Phonemes window at the bottom of the screen.

---

speech	sp"EC
database	d"At&b.As
phone_number	f"On_n"AmBR
address	&dr"es
script	skr"ipt

---

**Figure 4-7. Example of a Wordlist File with Phonetic Spelling**

### 4. Enter the following command to invoke the FlexWord Editor:

`wl_edit [-l chan#] [-s sp#] [-D dirname] [-O]` where,

- `-l chan#`

This parameter is the T/R channel #, normally 0, that is to be used for speech playback. If you do not specify this parameter, playback defaults to channel 0.

- `-s sp#`

This parameter is the SP circuit card # that is to be used for speech playback. SP circuit cards are referenced by an O.S. index number. If you do not specify this parameter, playback defaults to the SP circuit card with O.S. index 0.

- `-D dirname`

This parameter specifies the directory in which the FlexWord Editor starts.

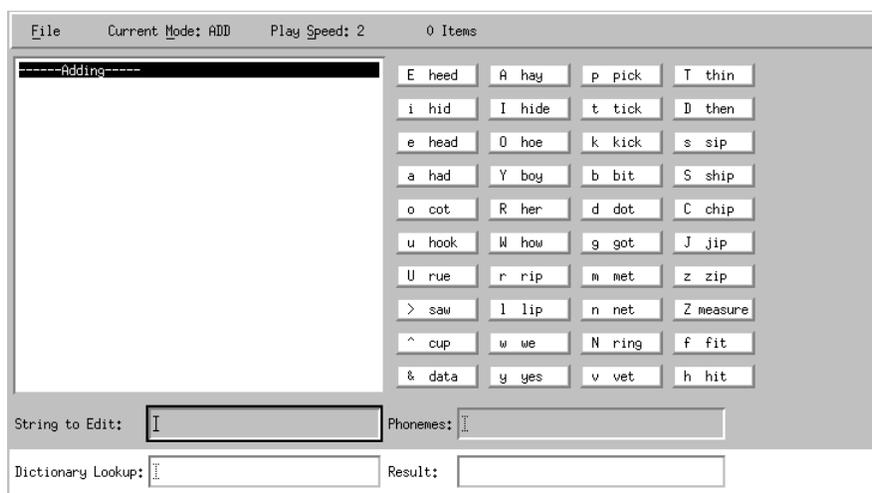
- `-O`

This parameter gives the maximum messaging output to the screen.

The FlexWord Editor allows you to hear and change the pronunciation for each of the words in the file and appears as shown in Figure 4-8.

**⇒ NOTE:**  
 If the Voice System is running, you are prompted to stop the VIS. See *Intuity CONVERSANT VIS V5.0 Maintenance*, 585-310-153, for information about stopping the voice system.

**⇒ NOTE:**  
 Error messages are written to a small window at the bottom of the screen. Messages are also logged to the file **/usr/tmp/wledit.output**



**Figure 4-8. FlexWord Editor**

5. Call into the T/R channel number you previously specified, if you call into a phone line.
6. Select File from the menu bar at the top of the FlexWord Editor.
7. Select Open from the File menu.
8. Highlight the wordlist file you created from the Files listing.

**⇒ NOTE:**  
 The directory that appears is the last directory you were working in before invoking the FlexWord Editor; therefore, it may not be the directory in which you created your wordlist file. You may need to select the directory in which the wordlist was created.

9. Select Open from the File menu. An example of the Open File menu appears as shown in Figure 4-9.



**Figure 4-9. Open File Menu in the FlexWord Editor**



**NOTE:**

Upon opening your file, the first word or phrase in your file is pronounced with TTS.

10. Listen to the phonetic pronunciation of each word by selecting each word in the wordlist.



**NOTE:**

Even if the word sounds correct, you may want to use the Wordlist Editor to change the word to reflect regional dialects. Or you may want to include two or more versions of the same word or phrase to account for different dialects.

11. Make sure to indicate any necessary changes to the phonetic spelling of your words/phrases in the Phonemes window. You can use the following emphasis marks as well as the phonemes shown in Figure 4-8.

- “ (quote)

Used to indicate primary emphasis. This emphasis mark must go before a vowel sound.

- . (period)

Used to indicate secondary emphasis. This emphasis mark must go before a vowel sound.

- \_ (underscore)

Used to separate words in a phrase

The emphasis mark must go before a vowel.

12. Select Save from the File menu to save the changes you made to your wordlist. An example of the Save menu appears as shown in Figure 4-10.



**Figure 4-10. Save File Menu in the FlexWord Editor**

13. Select Quit from the File menu to quit working in the FlexWord Editor.

## Adding Words and Phrases to a Vocabulary

Once you have created your wordlist, you can add words or phrases to the wordlist with the FlexWord Editor. To add a word or phrase to your file, perform the following procedure from the FlexWord Editor:

1. Select Open from the File menu.
2. Choose your wordlist file from the Files directory.
3. Select Open.

 **NOTE:**

Upon opening your file, the first word or phrase in your file is pronounced with TTS.

4. Decide where you want the new word or phrase to be placed in your file. The new word or phrase will be added below the currently selected word or phrase.
5. Select Add from the Current Mode menu.
6. Type the word or phrase to be added in the String to Add window.
7. Press **ENTER** to hear the pronunciation.

After pressing **ENTER**, the dictionary looks up the new word and adds it to the wordlist, while consecutively, Text-to-Speech gives a phonetic pronunciation of the word.

 **NOTE:**

TTS gives a pronunciation of the word or phrase according to what the dictionary lookup tells TTS. If the word or phrase is not pronounced correctly (for example, "live" has two pronunciations), you can change the phoneme string so that TTS pronounces it the way you want it. However, upon subsequent lookups of the word or phrase, TTS will still pronounce it according to the dictionary look up, because TTS does not deal with phonemes. You have to manually adjust the way in which TTS pronounces the word in the Phonemes window.

You can also listen to the pronunciations of each phoneme by clicking on the phonemes as shown in Figure 4-8.

8. Make sure to indicate any necessary changes to the phonetic spelling of your words/phrases in the Phonemes window. You can use the following emphasis marks as well as the phonemes shown in Figure 4-8.

- “ (quote)

Used to indicate primary emphasis. This emphasis mark must go before a vowel sound.

- . (period)

Used to indicate secondary emphasis. This emphasis mark must go before a vowel sound.

- \_ (underscore)

Used to separate words in a phrase

The emphasis mark must go before a vowel.

9. Press **ENTER** again to add the word or phrase to the list.

[10.] Go to step 3 if more words or phrases need to be added to the wordlist.

### **Deleting Words and Wordlists from a Vocabulary**

To delete a word or phrase from your file, perform the following procedure:

1. Access the FlexWord Editor by entering **wl\_edit**
2. Select Edit from the Current Mode menu.
3. Highlight the word or phrase you want to delete.
4. Select Delete from the Current Mode menu.

The designated word or phrase is removed from the file.

#### **⇒ NOTE:**

You must select Edit before selecting Delete. If you try to delete a word or phrase while in the Add mode, you will get the following message:

Cannot DELETE while in ADD mode!

## Copying Wordlists

To copy a wordlist to a target machine with FlexWord Speech Recognition loaded, perform the following procedure:

1. Move to the directory where your files are located.
2. Enter **wl\_copy** *<filename>* *<filename>* *<filename>* ...

The **wl\_copy** command copies the files or directories out to floppy disk. Names should be relative pathnames, not absolute pathnames, since they will be used to load the vocabularies onto a FlexWord system.

 **NOTE:**

If any of the wordlist names are directory names, the contents of the directories are also copied to floppy disk.

3. Take the disk to a machine with FlexWord Speech Recognition installed.
4. Run **wl\_install** to copy the wordlists from your disk to the designated machine.

 **NOTE:**

You do not have to use a floppy disk to copy wordlists; if you have FlexWord installed on the same machine, you can copy your wordlists to active and/or inactive directories. Refer to the section, "Installing Wordlists" for the correct directories for active and inactive wordlists.

## **FlexWord Vocabulary Administration**

Vocabulary administration basically involves moving wordlists between the active and inactive directories. However, prior to activating and deactivating a wordlist, the wordlist must be installed. This section includes how to install ("Installing Wordlists"), activate ("Activating Wordlists"), deactivate ("Deactivating Wordlists") and remove wordlists ("Removing Wordlists").

### **Installing Wordlists**

To install a wordlist onto the voice system, perform the following procedure:

1. Enter **wl\_install**
2. Insert the floppy disk containing the wordlists into the drive, when prompted. The wordlists are copied onto the floppy disk with the **wl\_copy** command.

The wordlists will be copied to **/att/asr/wordlists/inactive**. You will then be prompted as to whether any of the wordlists should be activated.

3. Designate, if appropriate, the wordlists that should be activated.

The designated wordlists are copied to **/att/asr/wordlists/active**.

The system creates the data files needed for FlexWord recognition.



#### **NOTE:**

The **/att/asr/wordlists/active** directory should contain only wordlists. A format check is done when the data files are generated, and, if any files other than wordlists are located in the **att/asr/wordlists/active** directory, an error message appears.

## Activating Wordlists

Activating a wordlist means creating all of the data files necessary to perform a FlexWord recognition. To activate a new wordlist, perform the following procedure:

1. Link (**ln**) all of the wordlists for the vocabulary from **att/asr/wordlists/inactive** into **/att/asr/wordlists/active**.

For example: **ln filename att/asr/wordlists/active**  
where *filename* is the name of the wordlist that you want to activate.

2. Enter **wl\_gen**

The **wl\_gen** command verifies the format of all active wordlists and then creates the data files needed for FlexWord recognition. If more than 38 phonemes, including the underscore (\_), are found with any word or phrase within a wordlist by the **wl\_gen** command, you receive an error message. Once **wl\_gen** finds an error within a wordlist, it quits looking at that wordlist, and the wordlist containing that word or phrase is not used when generating the FlexWord data files. Therefore, if you receive an error, you may have to run **wl\_gen** several times to iteratively locate each error. For more information about the **wl\_gen** command, see *Intuity CONVERSANT Voice Information System Version 5.0 Command Reference*, 585-310-230.



### CAUTION:

*The Voice System can be up when you are activating and deactivating vocabularies, but you should not run ScriptBuilder at the same time as the **wl\_gen** command.*

3. Diagnose the FlexWord SP circuit card after generating your wordlists. Refer to the *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230, for more information about the **diagnose card** command.
4. Verify and install the application associated with the active wordlist before you call up the application.

Refer to *Intuity CONVERSANT Voice Information System Version 5.0 Script Builder*, 585-310-727, for information about verifying and installing your Script Builder application.

## Deactivating Wordlists

Deactivating a wordlist means removing all of the data files associated with a particular vocabulary. To deactivate a wordlist, perform the following procedure:

1. Make sure you are in the **att/asr/wordlists/active** directory.
2. Remove all of the wordlists from **att/asr/wordlists/active**.

For example: **rm filename**, where *filename* is the name of the wordlist that you want to deactivate.

3. Enter **wl\_gen**

For more information about the **wl\_gen** command, see *Intuity CONVERSANT Voice Information System Version 5.0 Command Reference*, 585-310-230.

4. Diagnose the FlexWord SP circuit card. Refer to the *Intuity CONVERSANT VIS V5.0 Command Reference*, 585-310-230, for more information about the **diagnose card** command.

### Removing Wordlists

Removing a wordlist involves deleting both the data files and wordlists from the system. To remove a wordlist, perform the following procedure:

1. Deactivate the wordlist. See the instructions in the previous section, "Deactivating Wordlists".
2. Remove the associated wordlists from **/att/asr/wordlists/inactive**.

---

## **FlexWord Speech Recognition Accuracy**

---

Given the variance in human speech and the statistical properties of recognition algorithms, the speech recognizer may make occasional errors. FlexWord Speech Recognition accuracy depends not only on the recognition algorithms, but also on prompts, calling populations, the words to be recognized, and application designs.

Factors which positively influence accuracy rates include effective wordlists and prompt construction, as well as educated calling populations. Low recognition accuracy rates are usually caused by inexperienced callers who speak extraneous phrases and/or speak before the prompt is finished and wordlists that contain short words, rhyming words, or a large number of words.

### **Positive Influences on Recognition Accuracy**

The following items factor into positive recognition accuracy:

- **Effective wordlists**

Choosing words for your wordlist that have different sounds and are of medium length will help to increase recognition accuracy. For example, using the last name and the first name on wordlists of a name dialer application (applications that allow employees to speak a colleagues name rather than enter an extension number) increases recognition accuracy. In other types of applications, syllabic and vowel similarities may contribute to recognition confusion. For example, “women’s wear” and “men’s clothing” are more effective wordlist phrases than “women’s clothing” and “men’s clothing.” The latter pair sound too much alike; both have “men” embedded in the word and both share the word “clothing”.

Pay special attention to the words your customers actually use when they ask for a service. If you are automating an existing transaction, which has previously taken place between a customer and an agent, use the agent as a resource and try to mimic customers’ request words.

- **Experienced calling population**

Recognition improves for applications in which the calling populations are closed, and the callers are experienced with and/or trained to interact with the application.

- **Prompt structure and design**

Prompts offered in calm, clear voices greatly affect recognition accuracy as do the specific structures of the prompts. Prompts should guide the caller to say desired words/phrases.

- Prompts that are set up to dissuade callers from barging-in or speaking before the prompt is finished increase recognition accuracy. For example:

“Please say the month of your birth, now.”

— Menu prompts

For best results, menu prompts should be built with the structure:  
<desired result> <action required>. For example:

“To hear your checking account balance, say ‘checking.’

To hear your savings account balance, say ‘savings.’”

By placing the required action at the end of the prompt, the caller is able to remember the <action required> of her or him. Do not list the <action> before the <desired result>. For example, the prompt “Say ‘one’ for a description of the up-coming gallery events,” is a bad prompt. It encourages the caller to forget the specific <action required>, since the last thing related is the <desired result>.

■ Confirm

Overall accuracy can be increased if the application includes confirmation and reprompt steps. It is always a good idea to verify the recognized result *before* continuing with the application. For example:

“You said ‘swordfish.’ Is this correct? Please say ‘yes’ or ‘no.’”

Thus, the caller can make sure that her/his word matched the recognized response. See Figure 4-11 for an illustration of a confirmation path.

■ Informative prompts

Lengthy prompts providing detailed response instructions may improve recognition accuracy. But generally, from the perspective of frequent users, these types of prompts are not optional, since experienced users will have to listen to the complete prompt before responding. One solution may be to provide more informative prompts for first-time callers only. However, for applications with infrequent users, lengthy prompts may be more acceptable and help improve the accuracy.

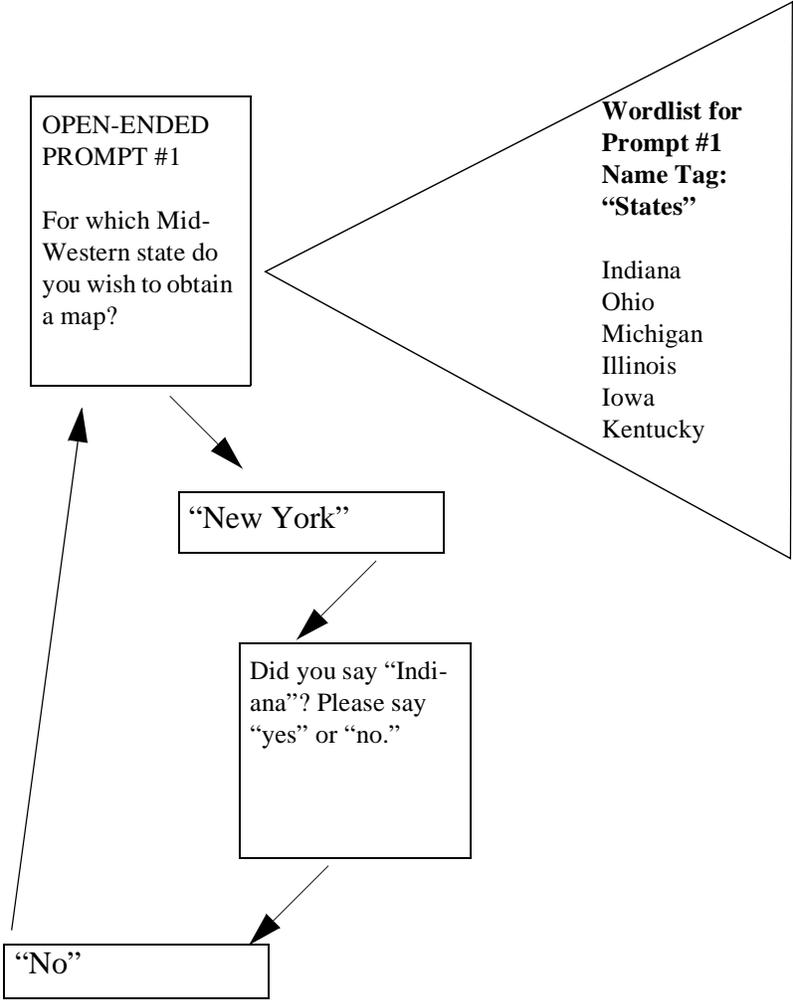


Figure 4-11. FlexWord Application Example of a Reprompt

## Negative Influences on Recognition Accuracy

The following items have a negative influence on recognition accuracy:

- Barging-In before the prompt is finished

FlexWord recognition does not support barge-in capabilities. Thus, prompts should include some sort of time reference so that callers know when to respond. The first prompt example shows how recognition accuracy can be increased by including a time reference:

PROMPT: "Please say the name of the agent with whom you wish to speak, now."

This prompt encourages the caller to wait until the prompt is finished before responding.

PROMPT: "Please say the name of the agent with whom you wish to speak."

The second prompt example does not have the time reference "now." Without the "now" at the end of the sentence, the caller may speak before FlexWord is ready.

 **NOTE:**

If an application is using FlexWord, you must specify "no" at the "Speak with Interrupt" on the Prompt and Collect screen.

- Environment

Noisy environments, such as an airport, train yard, or an unclear cellular phone connection may contribute to recognition accuracy problems.

- Ineffective wordlists

Ineffective wordlists cause lower recognition accuracy. In general, the larger the wordlist, the lower the expected recognition accuracy. The following are examples of problems in ineffective wordlists:

- Short words

One-syllable words that have the same vowel sounds are more difficult to recognize. For example, “on” and “off” both share the short “o” sound. (Long “o” vowel sounds appear in words like “oh” and “no” — words in which you can hear the letter “o.”) Short words should be used in moderation.

- Rhyming words

Wordlists that include words with similar rhyming vowel sounds can cause a decrease in recognition accuracy. Refer to the examples in Table 4-5 and make sure that you choose distinct sounding words when creating wordlists. Again, the VIS matches sounds and strings these sounds together. Pronounce the words on your wordlists and vary the vowel sound of each word if possible.

**Table 4-5. Examples of Vowel Sounds**

Long “A”	Short “A”	Long “E”	Short “E”	Long “I”	Short “I”	Long “O”	Short “O”	Long “U”	Short “U”
paper	fatten	peaceful	metal	spicy	trigger	postal	foggy	cupid	grunting
traitor	dancer	receipt	header	tiger	kitty	soda	robber	viewer	runner

- Wordlist size

As the size of wordlists increase, accuracy decreases. The best accuracy results can be achieved by structuring an application to make use of several smaller wordlists rather than one large wordlist.

## **WholeWord Versus FlexWord Accuracy**

---

For the Speech Recognition feature, accuracy varies based upon the recognition method and application in the following order, with the first list item giving the highest accuracy, the second item giving the next best accuracy, and so on.

1. WholeWord American English Isolated Word
2. WholeWord Connected Word
3. FlexWord Isolated Word

For a WholeWord string of digits, the per-digit accuracy is comparable to isolated word recognition. However, the accuracy of the whole string is lower than the per-digit accuracy, and steadily decreases as more digits are added. It is recommended that digit strings be limited to 10 digits or less to achieve optimum accuracy.

---

## Putting It Together

# 5

---

### What's in This Chapter

This chapter discusses how to get the most out of your speech-related features, why certain features work well together, and when you can use them together.

## **Getting the Most Out of WholeWord**

---

The best way to provide service to callers who do not have touch-tone service or who do not want to use touch-tone service is to allow callers to speak their responses. WholeWord Speech Recognition can be used to recognize a limited set of words, as discussed in Chapter 4, "Recognizing Speech Input". However, WholeWord Speech Recognition is most successful when you use it to augment a touch-tone application by handling callers who do not have touch-tone telephones.

WholeWord Speech Recognition is the only way to provide connected-digit recognition. If your application requires recognition of strings of digits, then you must use WholeWord Speech Recognition.

## **Combining WholeWord Speech Recognition and Touch Tone**

---

In order to handle callers who do not have touch-tone telephones, you can use WholeWord Speech Recognition to allow rotary callers to speak their response. For information on how to incorporate WholeWord Speech Recognition into your touch-tone application, see *Intuity CONVERSANT VIS V5.0 Application Design Handbook*, 585-310-551.

---

## Getting the Most Out of FlexWord

---

As discussed in Chapter 4, "Recognizing Speech Input", FlexWord recognizes callers speaking words from a vocabulary that you define, and therefore is specifically tailored to your application. Allowing the caller to speak the option they want rather than speaking a number assigned to the option can make the interaction more natural and easy to use. This section offers guidelines for getting the most out of FlexWord Speech Recognition.

### Building a FlexWord Vocabulary

---

As discussed in Chapter 4, "Recognizing Speech Input", you can build FlexWord vocabularies two ways:

- Use the optional FlexWord Toolkit to make your own wordlists quickly and easily. An advantage to using the FlexWord Toolkit is that you can use the wordlists in your application immediately after you define them.
- Have AT&T build your vocabularies through the AT&T Custom Vocabulary service

 **NOTE:**

Consider the size of your vocabularies, the number of FlexWord application scripts you plan to have, and how much your vocabulary may change over time to help you decide whether or not to purchase the FlexWord Toolkit or the Custom Vocabulary service.

### Choosing a FlexWord Vocabulary

---

Since FlexWord offers you the freedom to specify your own custom vocabularies, you have the task of making sure your application is easy to use. How you set up your wordlists contributes greatly to the success of your application.

FlexWord vocabulary items can be single words or phrases, which are all referred to as "words." The words must be divided into groups called wordlists. A wordlist includes all of the words that can be spoken at a particular prompt. Each application has its own vocabulary, or set of wordlists. And, at any prompt, only one wordlist can be active at a time. This means that only one wordlist can be under consideration by the recognizer at any one time. The words in each wordlist must be distinct enough from each other to allow the recognizer to work effectively. It doesn't matter if two words are very similar, as long as they are on two separate wordlists. The following sections discuss factors that you should consider when creating your wordlists.

## **Caller Error**

---

You may find that callers tend to respond with an unacceptable word or phrase during a prompt. If so, you can add these words to the wordlist. In your application, you can handle these words as caller errors and reprompt the caller. For example, if your wordlist contains account names and the valid account names are “checking” and “savings,” you could add “loan” to the wordlist. If the recognizer recognizes that a caller says “loan,” you can design your application to say “Sorry. We did not recognize your response. Please choose ‘checking’ or ‘savings.’ “

## **Word Length**

---

Words with more than one syllable are recognized more reliably than shorter words. For example, the phrase “add entry” would be better to use than the word “add.” In fact, phrases can even be fairly long, although the longer the phrase is, the more of a chance there is of callers forgetting it or making mistakes while speaking.

## **Word Meaning**

---

Choose words and phrases that would occur naturally to your typical caller. One way to determine this is to survey some representative callers and ask what words or phrases they would use for actions in your script. Try not to rely on just your knowledge, because your impressions may be very different than those of your callers.

## **Prompting for Input**

---

Because FlexWord does not support barge-in, you should structure your prompts to encourage callers to wait until the end of the prompt before speaking. For example, “Please enter the last four digits of your social security card, now,” as opposed to “Now enter the last four digits of your social security card, please.”

## **Sound of a Word**

---

Try to use words in your wordlists that do not sound alike. To the recognizer, vowels are slightly more important than consonants. Also, avoid using words with similar vowel sounds. Words that differ in only one or two consonants are difficult for the recognizer to distinguish between. For example, the words “connect” and “comment” sound similar to the recognizer.

## **Synonyms**

---

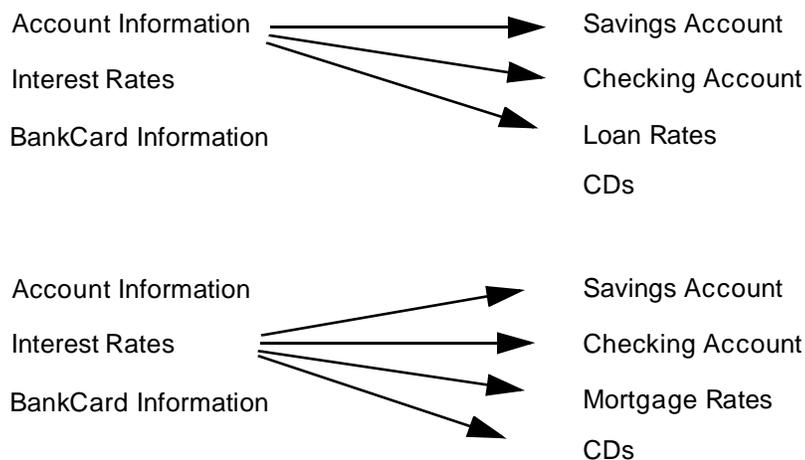
If you find that callers use more than one word for the same thing, you may want to add both words to your list.

## Examples of FlexWord Applications

This section includes two examples of suggested target FlexWord Speech Recognition applications.

### Menu-Based Applications

Menu-based applications that use words and phrases native to your business environment are suggested to be used with FlexWord. Your application flows smoothly, using words that would have occurred naturally in an interaction between an agent and the customer. For example, refer to Figure 5-1, for an example of a menu-based application.



**Figure 5-1. Example of Menu-Based Application**

### Applications Requiring Large Wordlists

Applications that require large, custom wordlists would benefit from FlexWord Speech Recognition. In this type of application, the voice prompt could not enumerate all of the valid responses. Because of this, the application must provide sufficient guidance to the caller to facilitate an appropriate response. An example of this type of application would be a name-dialer.

## **Getting the Most out of Text-to-Speech**

This section offers several tips to consider when using Text-to-Speech (TTS) in prompts and announcements.

When Text-to-Speech prompts are used with touchtone input, callers can interrupt the prompt just as they can when the prompt has been prerecorded. Callers can also interrupt TTS prompts with WholeWord input. However, when using TTS prompts with FlexWord input, barge in is not available. Therefore, when accepting FlexWord input, structure your TTS prompts so that they encourage callers to wait until the end before responding.

**⇒ NOTE:**

The same information discussed above also applies to using TTS in announcements.

## **Using FlexWord Speech Recognition and Text-to-Speech**

---

Since FlexWord Speech Recognition provides the ability to recognize up to 2000 words, your FlexWord application needs a way to speak out the recognized responses. This can be done by recording all the words in your wordlist, and speaking them back after a database lookup.

However, TTS provides a much more flexible way of speaking the wordlist. TTS allows you to speak the contents of `$CI_VALUE` (the recognized word on the wordlist) very easily. Also, if the wordlist changes, TTS still works — possibly without changes to your application. If you use prerecorded speech to speak the wordlist, each time the wordlist changes, the new words will need to be recorded. TTS eliminates this step.

## Using Text-to-Speech and Prerecorded Speech

A single script can speak prompts and announcements in prerecorded speech only, Text-to-Speech (TTS) only, or a combination of both. Below is an example of when you could use both in your script.

In this example, recorded speech is used for all phrases except for speaking the customer name. TTS allows us to speak the contents of the "cust\_name" variable, which contains the customer's name. The speaking of the actual customer's name would be difficult to do without TTS. You would have to record every possible customer name, save the phrase tag numbers of each customer's name, and associate the phrase tag number with the customer account number. Then you would speak (using NX format) the phrase tag number corresponding to the customer's account number.

```
1.  Prompt & Collect
    Prompt
    Speak With Interrupt
    Phrase: "Please say your 6 digit account number."
    Input
    Caller Input Field: account
    Mode: US_DIG
    Min Number Of Digits: 06
    Max Number Of Digits: 06
    Checklist
    Case: "nnnnnn"
    Speak With Interrupt
    Phrase: "Thank you. Please hold while we
           verify acct. num."
    Continue
    Case: "Not On List"
    Reprompt
    Case: "Initial Timeout"
    Reprompt
    Case: "Too Few Digits"
    Reprompt
    Case: "No More Tries"
    Quit
    End Prompt & Collect
2.  Read Table
    Table Name: ACCOUNTS      Search From Beginning
    Field: account_num = account
3.  Prompt & Collect
    Prompt
    Speak With Interrupt
    Phrase: "Your name is"
    Field: cust_name As A
    Phrase: "Please say yes or no."
    Input
    Mode: US_YN
    Max Number Of Digits: 01
```

Checklist

Case: "Input Ok"

Continue

Case: "Initial Timeout"

Reprompt

Case: "Too Few Digits"

Reprompt

Case: "No More Tries"

Quit

End Prompt & Collect

## Using WholeWord and FlexWord Together

---

FlexWord and WholeWord speech recognition can be used in a single application script. This allows you many possibilities and flexibility in what you can ask callers to say. When using Script Builder, you can specify that a Prompt and Collect action step use either FlexWord Speech Recognition or WholeWord Speech Recognition, depending on what you want to prompt callers to say. If you want callers to say “yes” or “no,” a series of digits, or single digits, you would choose a WholeWord recognition type. If you want callers to speak a word or phrase from your custom vocabulary, you would choose a FlexWord wordlist.

However, each Prompt and Collect action must specify FlexWord or WholeWord. Both packages can not be performing in the same prompt. To see a model of a retail application example which combines FlexWord’s wordlists and WholeWord’s “yes/no” speech recognition feature, see Figure 5-2.

In Figure 5-2, the color “blue” must appear on both the “interior color” wordlist and the “exterior color” wordlist. Since the word “blue” appears on two wordlists, it counts as 2 words. If the word “blue” is not on the “interior color” wordlist, this feature *will not* look to the “exterior color” wordlist to find it. The feature will return the word from the wordlist that it thinks is the most suitable match. In that the feature can only look at one wordlist per Prompt and Collect screen, it is of the utmost importance that each wordlist contains *all* possible choices for a single prompt.

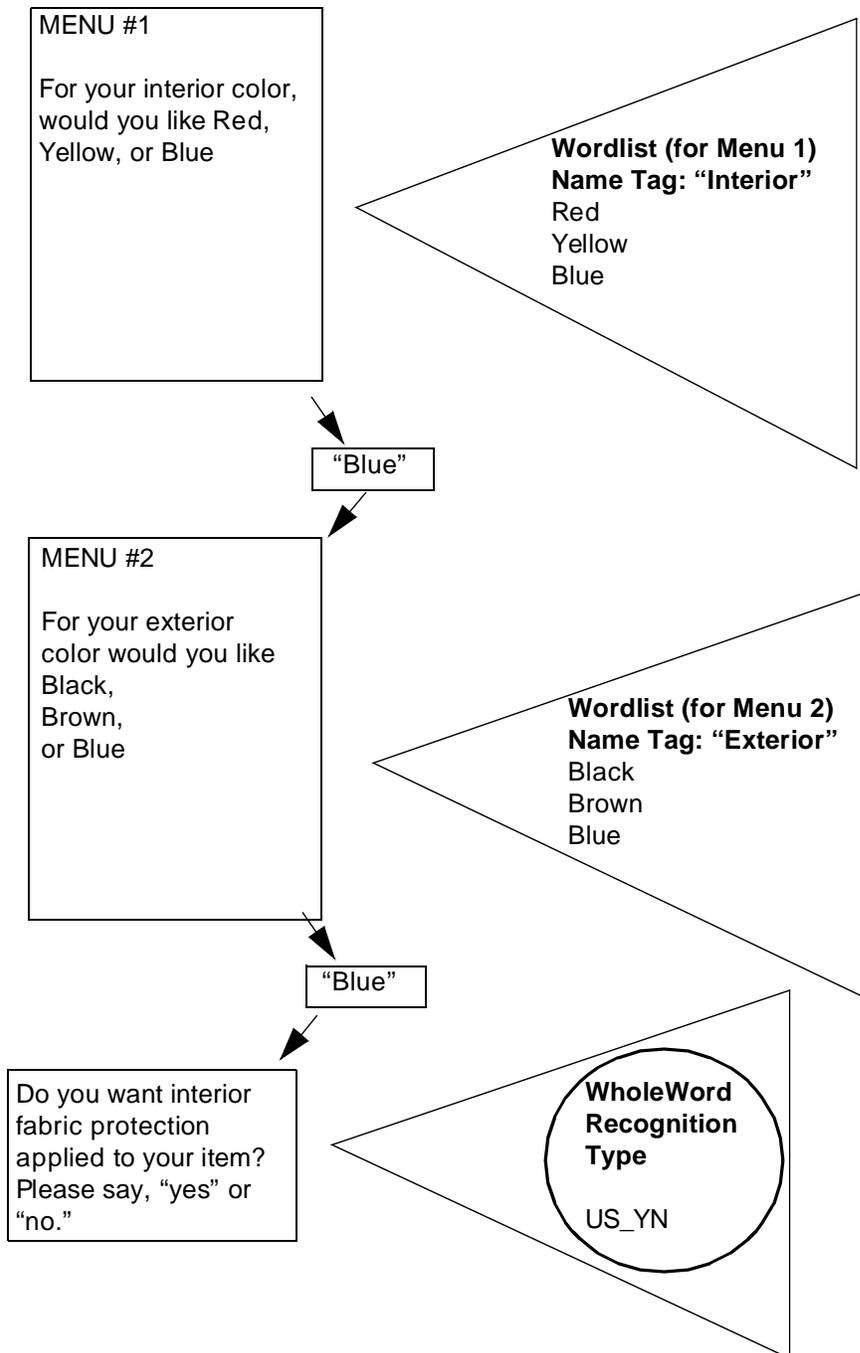


Figure 5-2. Example with FlexWord and WholeWord

## **WholeWord versus FlexWord**

---

This section illustrates the differences between WholeWord and FlexWord Speech Recognition (Table 5-1).

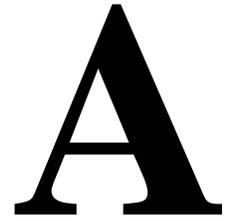
**Table 5-1. WholeWord versus FlexWord Speech Recognition**

<b>WholeWord</b>	<b>FlexWord</b>
Word-based	Phoneme-based
Requires data collection*	No data collection required
Connected word	Isolated word/phrase
Standard grammars	Customized wordlists
Barge-in supported	Barge-in not supported
Phrase screening supported	Phrase screening supported
Limited vocabulary	2000+ word vocabulary
Word spotting supported	Word spotting supported

\*WholeWord Speech Recognition requires data collection if you want words other than “zero” through “nine,” “yes,” “no,” and “oh.”

---

## **Standard Speech Phrases**



---

### **What's in This Appendix**

This appendix provides a complete listing of the phrases that are included in the Standard Speech package.

## **Standard Speech Phrases**

---

This section provides a complete listing of all standard speech phrases. Three types of inflection exist with speech phrases:

- Rising inflection is usually used in questions and at the beginning of some words. Rising inflection is denoted in Table A-1 with a question mark (?) that follows the tag.
- Medial inflection is usually used in the middle of a word or statement. For example, when you speak the number "101," the "0" is spoken with medial inflection. In Table A-1, tags that are followed neither by a question mark nor a period indicate medial inflection.
- Falling inflection is usually used at the end of a word or statement. For example, when you speak "2.0," the "0" is spoken with falling inflection. Falling inflection is denoted in Table A-1 with a period (.) following the tag.

**Table A-1. Standard Speech Phrases**

:a	:20	:9	:x?
:b	:30	:10.	:y?
:c	:40	:11.	:z?
:d	:50	:12.	:0?
:e	:60	:13.	:1?
:f	:70	:14.	:2?
:g	:80	:15.	:3?
:h	:90	:16.	:4?
:i	:100	:17.	:5?
:j	:1000	:18.	:6?
:k	:1000000	:19.	:7?
:l	:a.	:20.	:8?
:m	:b.	:30.	:9?
:n	:c.	:40.	:10?
:o	:d.	:50.	:11?
:p	:e.	:60.	:12?
:q	:f.	:70.	:13?
:r	:g.	:80.	:14?
:s	:h.	:90.	:15?
:t	:i.	:100.	:16?
:u	:j.	:1000.	:17?
:v	:k.	:1000000.	:18?
:w	:l.	:a?	:19?
:x	:m.	:b?	:20?
:y	:n.	:c?	:30?
:z	:o.	:d?	:40?
:0	:p.	:e?	:50?
:1	:q.	:f?	:60?
:2	:r.	:g?	:70?
:3	:s.	:h?	:80?
:4	:t.	:i?	:90?
:5	:u.	:j?	:100?
:6	:v.	:k?	:1000?
:7	:w.	:l?	:1000000?
:8	:x.	:m?	:am
:9	:y.	:n?	:April
:10	:z.	:o?	:April.
:11	:0.	:p?	:April?
:12	:1.	:q?	:August
:13	:2.	:r?	:August.
:14	:3.	:s?	:August?
:15	:4.	:t?	:cent
:16	:5.	:u?	:cents
:17	:6.	:v?	:December
:18	:7.	:w?	:December.

*Continued on next page*

**Table A-1. Standard Speech Phrases — Continued**

:19	:8.	:x?	:December?
:dollar	:Monday	:thirteenth.	:Wednesday.
:dollar and	:Monday.	:thirteenth?	:Wednesday?
:eighteenth	:Monday?	:thirtieth	
:eighteenth.	:ninth	:thirtieth.	
:eighteenth?	:ninth.	:thirtieth?	
:eighth	:ninth?	:thirtyfirst	
:eighth.	:noon	:thirtyfirst.	
:eighth?	:November	:twelfth	
:eleventh	:November.	:thirtyfirst?	
:eleventh.	:November?	:thursday	
:eleventh?	:o'clock	:thursday.	
:February	:October	:thursday?	
:February.	:October.	:tuesday	
:February?	:October?	:tuesday.	
:fifteenth	:oh (as in 8:05)	:tuesday?	
:fifteenth.	:pm	:twelfth.	
:fifteenth?	:point	:twelfth?	
:fifth	:Saturday	:twentieth	
:fifth.	:Saturday.	:twentieth.	
:fifth?	:Saturday?	:twentieth?	
:first	:second	:twentyeighth	
:first.	:second.	:twentyeighth.	
:first?	:second?	:twentyeighth?	
:fourteenth	:September	:twentyfifth	
:fourteenth.	:September.	:twentyfifth.	
:fourteenth?	:September?	:twentyfifth?	
:fourth	:seventeenth	:twentyfirst	
:fourth.	:seventeenth.	:twentyfirst.	
:fourth?	:seventeenth?	:twentyfirst?	
:Friday	:seventh	:twentyfourth	
:Friday.	:seventh.	:twentyfourth.	
:Friday?	:seventh?	:twentyfourth?	
:January	:sixteenth	:twentyninth	
:January.	:sixteenth.	:twentyninth.	
:January?	:sixteenth?	:twentyninth?	
:July	:sixth	:twentysecond	
:July.	:sixth.	:twentysecond.	
:July?	:sixth?	:twentysecond?	
:June	:Sunday	:twentyseventh	
:June.	:Sunday.	:twentyseventh.	
:June?	:Sunday?	:twentyseventh?	
:March	:tenth	:twentysixth	
:March.	:tenth.	:twentysixth.	

*Continued on next page*

**Table A-1. Standard Speech Phrases — Continued**

:March?	:tenth?	:twentysixth?	
:May	:third	:twentythird	
:May.	:third.	:twentythird.	
:May?	:third?	:twentythird?	
:midnight	:thirteenth	:Wednesday	



---

## **Speech File Formats**

# **B**

---

### **What's in This Appendix**

This appendix includes information about speech files and speech file formats.

## **Speech File Formats**

---

Before the speech phrases are digitized and then stored, the speech phrases must be in a supported format. This is the process of encoding speech.

Intuity CONVERSANT VIS supports the following encoding formats for speech phrases:

- Pulse Code Modulation (PCM) at 64 kbps in Mu-law encoding format
- Adaptive Differential Pulse Code Modulation (ADPCM) at 16 kbps
- ADPCM at 32 kbps
- Sub Band Coding (SBC) at 16 kbps (not commonly used in VIS applications)
- SBC at 24 kbps (not commonly used in VIS applications)

### **More About PCM Speech File Formats**

---

In a PCM speech file, speech is sampled at 8000 times a second. Each sample is digitally coded into an 8-bit pattern (allowing 256 levels), resulting in 64000 bits-per-second of speech. The header, which has the values *0xAA 0xFF 0x34 0x00*, is repeated at least five times for every second of speech.

The speech data consists of contiguous 8-bit patterns representing sampled speech. Depending on the coding technique used, two types of PCM formats are possible, linear and Mu-law.

#### **Linear PCM Format**

In the Linear PCM format, sound levels of successive digital encoding are equally spaced in the binary form. Some third party applications use the Linear PCM format.

#### **Mu-law PCM Format**

In the Mu-law PCM format, PCM coding uses predefined quasi logarithmic steps for speech levels, encoding more steps when the speech level is low. The VIS uses Mu-law encoding and is assigned the identifier code N as *0x34*.

### **More About ADPCM Speech File Formats**

---

ADPCM speech encoding is based on the principle that it is possible to reduce the amount of information needed to transmit speech between the sender and the receiver by using appropriate mathematical algorithms. The following types of ADPCM speech file formats are available:

- ADPCM-32
- ADPCM-16

Both of the formats follow similar compression algorithms.

Given that natural speech follows specific patterns (as opposed to random noise), the encoder (or the sender) predicts the present speech level by using a predefined algorithm and considering the past speech history. This predicted value is then compared to the actual present speech level and the difference is encoded into digital format. The difference between the actual and the predicted values can be made as small as possible by using suitably defined predictor algorithms.

Thus, it is possible to encode the difference between the predicted and the actual speech into digital samples of either 4 bits (16 levels) or 2 bits (4 levels). At the decoder (or the receiver) end, the process is reversed. The decoder uses an equivalent algorithm to predict the present speech level from the past history, and makes the correction based on the received information from the sender to get the actual speech level. The accuracy and performance of the speech encoding and decoding depends on the type of algorithm used as well as the number of bits used for digital encoding of the difference between predicted and actual speech.

- In the ADPCM 32 speech file, the header, which has the values *0xAA 0xFF 0x32 0x00*, is repeated at least five times for every second of speech data.
- In the ADPCM 16 speech file, the header, which has the values *0xAA 0xFF 0x30 0x00*, is repeated at least five times for every second of speech data.

If the value *0xAA 0xFF* occurs anywhere within the speech portion of the file, it is changed to *0xAA 0xFE*. This makes a small but imperceptible change to the speech.

Processing of speech data for an application may be accomplished concurrently with application development. The only requirement is that the digitized speech must be loaded on the system before the application can be assembled.

## More About Speech Files

Speech files are comprised of two parts:

- Header section of the format shown in Figure B-1 which is present at the beginning of the file and repeated at periodic intervals within the body of the file
- Encoded digital data representing speech



**Figure B-1. Header Section**

The following rules apply to the header section:

- *0xAA* and *0xFF* are the two bytes of data with bit patterns *01100110* and *11111111*
- *N* is a unique identifier code representing the speech encoding format as shown in Table B-1.
- *L* is a mandatory field that represents the length of a control field that follows the header bytes. The control field is optional. If it is absent, *L* is set to *0x00*. For the PCM and ADPCM type speech files, it is set to *0x00* (no control field is specified).

To encode *N*, values *0x40* or less are reserved for specific speech types. Values greater than *0x40* are user defined. If *N* is greater than *0x40*, the length field *L* identifies the number of words (two bytes each) that follow which include user defined information.

**Table B-1. Identifier Codes in Speech Encoding**

N	Value
PCM (Mu-law)	0 x 34
ADPCM 32	0 x 32
ADPCM 16	0 x 30
SBC 24	0 x 21
SBC 16	0 x 20
User defined	0 x 41 or greater

Header bytes are inserted into the speech file so that the header appears at least five times for every second of speech. The headers are aligned on even byte boundaries.



---

## **Computing Channel Numbers**

# **C**

---

### **What's in This Appendix**

This appendix includes the procedure for computing channel numbers for the GSE and the FlexWord Toolkit.

## Computing Channel Numbers

---

The Graphical Speech Editor command, **gse [-l<chan>] | [-p<chan>-r<chan>]**, requires the identification of a channel number (chan) for each respective channel designation (-l, -p, and/or -r). This channel number is not the VIS channel number on the tip-ring card, but a Channel O.S. index for the particular channel.

To determine the appropriate channel numbers for the l, p, and r channel designations on the Tip/Ring circuit card, perform the following procedure:

1. Identify the VIS channel that you want to use before accessing the system.  
For the sake of this exercise, let us assume #7 to be the identified channel.
2. Log on to the system by performing the following:
  - a. Enter root at the system prompt.  
The password prompt is displayed
  - b. Enter your valid password.  
The UNIX system prompt (#) will then appear.

3. Display the present channel settings on the tip-ring card by entering **disp ca tr**

The Tip/Ring channel settings are displayed, as shown in the example in Figure C-1.

---

Card 0 is IVP6	O.S.Index: 0	Function: TipRing
	State: Insert	Options: master2,tdml,tt
CD.PT CHN STATE	STATE-CHNG-TIME	SERVICE-NAME PHONE GROUP OPTS TYPE
0.0 0 Inserv	Sep 15 12:04:55	eshtest - 2 tdm IVP6
0.1 1 Foos	Sep 15 12:04:55	- - 2 tdm IVP6
0.2 2 Foos	Sep 15 12:04:55	- - 2 tdm IVP6
0.3 3 Foos	Sep 15 12:04:55	eshtest - 2 tdm IVP6
0.4 4 Foos	Sep 15 12:04:55	- - 2 tdm IVP6
0.5 5 Foos	Sep 15 12:04:55	- - 2 tdm IVP6
Card 1 is IVP6	O.S.Index: 2	Function: TipRing
	State: Broken	Options: master3,tdml,tt
CD.PT CHN STATE	STATE-CHNG-TIME	SERVICE-NAME PHONE GROUP OPTS TYPE
1.0 6 Broken	Sep 15 12:05:43	- - 2 tdm IVP6
1.1 7 Broken	Sep 15 12:05:43	- - 2 tdm IVP6
1.2 8 Broken	Sep 15 12:05:43	- - 2 tdm IVP6
1.3 9 Broken	Sep 15 12:05:43	- - 2 tdm IVP6
1.4 10 Broken	Sep 15 12:05:43	- - 2 tdm IVP6
1.5 11 Broken	Sep 15 12:05:43	- - 2 tdm IVP6

---

**Figure C-1. Example of Tip-Ring Channel Settings**

4. You must now find the O.S. index for the card on which the identified channel is located. In the example, Channel #7 (under CHN — second column) is located on Card 1 which has “2” as its O.S.Index number.
5. The next step is to find the channel offset. This number ranges from 0 through 5 and is the one digit suffix listed under CD.PT (first column). The offset for channel 7 is 1.
6. The O.S.Index for the channel can now be computed. The formula to be used in this computation is as follows:

Channel O.S.Index = 6 x (card's os index) + offset

In the example, the O.S.Index for channel 7 is  $6 \times 2 + 1 = 13$ .

For example, entering **gse -l13** sets up GSE so that you can call channel 7 to play and record.

This O.S.Index can now be used as an argument when invoking GSE.



---

## Advanced Text-to-Speech Features

# D

---

### What's in This Appendix

This appendix discusses several ways to customize synthesized speech, by including escape sequences to add silence delays, change the speaking rate, and mark text as belonging to a specific text category.

## Introduction

---

With some Text-To-Speech (TTS) applications, you may need to further customize the use of synthesized speech — for example, by adding silence delays, changing the speaking rate, or by marking text as members of a more specific text category. While you can alter the way in which TTS speaks a phrase, some applications require that you use escape sequences placed before, after, or within the spoken text.

Not all applications require these advanced features. If you have questions about the use of Advanced TTS features, call the Customer Support Hotline at 1-800-344-9670.



### **CAUTION:**

*Use the escape sequences exactly as they are described in this appendix. Using them differently could create problems in your application.*

## Using Escape Sequences

---

The manner in which TTS speaks a phrase can be altered by using specific escape sequences within the text. Several methods exist for adding escape sequences to your text. One method is to edit your text directly. Another method is to concatenate the specific escape sequence to your text string. Examples of these two methods are provided at the end of this section.

## Silence Delays

---

Several ways to introduce silent pauses or delays during the speaking of synthesized text are available. Most application silence needs can be handled by using the methods described in “Hints for Writing Applications Using Text-to-Speech” in Chapter 6, “Using Optional Features With Script Builder,” of *Intuity CONVERSANT VIS Version 5.0 Script Builder*, 585-310-727. However, if your application requires more advanced silence delays, you may use an escape sequence. With escape sequences, silence delays can be added to the beginning, end, or within a text phrase. The application developer indicates the silence duration in centiseconds (one hundredth of a second). One hundred centiseconds equals one second. The escape sequences shown in Table D-1 are used for silence delays.

**CAUTION:**

*Pay careful attention to the limits on the duration of silence delays. Using values outside of the limits could cause problems in your application.*

**Table D-1. Escape Sequences for Silence Delays**

<b>Escape Sequence</b>	<b>Definition</b>	<b>Example</b>
<code>\!si &lt; n &gt;</code>	For $n$ centiseconds of silence at the beginning of a phrase, where $1 < n < 100$ . This escape sequence can only be used at the beginning of a text line, as it is used in the example where .7 seconds of silence occurs before the word "Hello" is spoken.	Announce Speak with Interrupt Text: <code>"\!si70 Hello! Welcome to your Home Shopping Line!"</code>
<code>\!sf &lt; n &gt;</code>	For $n$ centiseconds of silence at the end of a phrase, where $1 < n < 60$ . This escape sequence can only be used at the beginning of a text line, as it is used in the example where 0.5 seconds of silence occurs after the first sentence is spoken but before the second sentence is spoken.	Announce Speak with Interrupt Text: <code>"\!sf50 Hello, welcome to your Home Shopping Line! Text: For information about this service, press 1. Text: To start your order, press 2."</code>
<code>\(* [&lt;n&gt;]\)</code>	For $n$ centiseconds of silence within a phrase, where $1 < n < 100$ .  In the example, 70 centiseconds of silence occurs after the word "Hello" is spoken and before the word "and" is spoken. The maximum allowance is 100 centiseconds.	Announce Speak with Interrupt Text: <code>"Hello \(*[70]\)</code> and Text: <code>welcome to your</code> Text: <code>Home Shopping Line!"</code>

## Speaking Rate

---

The rate of the synthesized voice can be controlled through the use of an escape sequence. Five rates are available, from 0 (quickest) to 4 (slowest). The default rate is 2. Each rate setting is effective until the end of the sentence (text ending with punctuation such as a period, question mark, or exclamation mark). To insure that the rate returns to the default, you must manually change the rate back to 2. The escape sequence shown in Table D-2 are used to change the speaking rate.

**Table D-2. Escape Sequences for Speaking Rate**

Escape Sequence	Definition	Example
\!r < n >	<p>Where <i>n</i> is the new rate value between 0 and 4.</p> <p>In the example, the slower rate is in effect only while the order number is spoken. The first sentence, up to the rate change, and the second sentence is spoken with the default speaking rate.</p>	<p>Announce            Speak with Interrupt            Text: "Your order number is                  \!r4 104235\!r2 ."            Text: Please use this number                  when checking on your                  order."</p>

## Additional Text Classes

With the **Prompt & Collect** and **Announce** action steps, you can indicate the format of the text to be spoken. The text can be marked to be spoken in the following formats:

- Character
- Number
- Time
- Date
- Dollar amounts

Some applications may require a further classification of the text to be spoken. The following additional escape sequences are available for further classifications:

- Addresses
- Telephone numbers
- Fractions
- Proper names.

To speak text from these additional class types, you must activate a *class detector* for a specific text class. The detector can be set to one of the following three detection modes:

- Off
- Conservative
- Risky

Initially, all the class detectors are set in the conservative mode. Class detector escape sequences do not need to be embedded in or connected to some other text to be activated. The mode can be changed by using an “Announce” action step and speaking only the escape sequence. The class detector will remain in the new state until it is changed or until the script ends.

### **NOTE:**

Only use Risky mode when the spoken text is known to be a member of the specified text class (for example, telephone numbers) or classes (for example, names and addresses).

## Changing Class Detection

Text class detection modes can be changed using escape sequences with the following general format:

```
\!n < c >< m >
```

The *c* variable is one of the following text classes:

- **a** (Address)  
Use risky mode to detect partial addresses and to correctly pronounce ambiguous address abbreviations, such as “Dr.” for “Drive.”
- **f** (Fraction)  
Use risky mode so that, for example, “1/4” is pronounced as “one fourth.”
- **n** (Proper name)  
Use risky mode so that titles such as “Mr.” for “Mister” or ambiguous proper name abbreviations such as “Dr.” for “Doctor” are pronounced correctly.
- **p** (Telephone numbers)  
Use risky mode to ensure that vanity telephone numbers such as 1 (800) CALL ATT are spelled out.

The *m* variable is one of the three possible class detector mode settings for the specified text class:

- **o** (Turn off the class detector)
- **c** (Set the class detector to conservative mode — default)
- **r** (Set the class detector to risky mode)

## Setting Specific Class Detector Modes

This section includes examples of setting specific class detector modes.

- **Proper Name Detector**  

```
\!nnr (Set the Proper Name Detector to Risky Mode)  
\!nnc (Set the Proper Name Detector to Conservative Mode — default)  
\!nno (Set the Proper Name Detector to Off)
```

The example that follows shows setting the proper name detector to risky mode, speaking a customer name from a database field, and then setting the proper name detector back to conservative mode. The risky mode setting is needed only if your data is known to have name-related abbreviations or titles. Otherwise, use the default mode.

```
Announce  
  Speak with Interrupt  
  Text:  "\!nnr"
```

```
Text: "Your name is"  
Field: customer_name as A  
Text: "\!nnc"
```

- Telephone Number Detector

\!npr (Set the Telephone Detector to Risky Mode)

\!npc (Set the Telephone Detector to Conservative Mode — default)

\!npc (Set the Telephone Number Detector to Off)

The example that follows shows setting the telephone number detector to risky mode, speaking a vanity phone number, and turning the telephone number detector off because no more phone numbers are expected.

```
Announce  
Speak with Interrupt  
Text: "Please call \!npr" 1-800 SHOPPER \!npo  
Text: for more information."
```

- Address Detector

\!nar (Set the Address Detector to Risky Mode)

\!nac (Set the Address Detector to Conservative Mode — default)

\!nao (Set the Address Detector to Off)

The example that follows shows setting the address detector to risky mode, speaking out an address from a database field, and then turning the address detector to conservative mode.

```
Announce  
Speak with Interrupt  
Text: "\!nar"  
Text: "Your address is"  
Field: customer_address as A  
Field: customer_state as A  
Text: "\!nac"
```

## **Sample Application Using Escape Sequences**

The following is a sample Script Builder application that uses escape sequences.

```
start:
1.  Answer Phone
2.  Set Field Value
    Field: ADDRESS_RISKY_MODE = "\!nar"
    Field: ADDRESS_DEFAULT_MODE = "\!nac"
    Field: PHONE_RISKY_MODE = "\!npr"
    Field: PHONE_DEFAULT_MODE = "\!npc"
    Field: NAME_RISKY_MODE = "\!nnr"
    Field: NAME_DEFAULT_MODE = "\!nnc"
    Field: BEG_SILENCE = "\!si70"
    Field: MIDDLE_SILENCE = "\!(*[50])"
    Field: END_SILENCE = "\!sf70"
    Field: SPEAKING_RATE_FOUR = "\!r4"
3.  Announce
    Speak With Interrupt
    Text: "Hello! Welcome to your home shopping line!"
4.  Prompt & Collect
    Prompt
    Speak With Interrupt
    Text: "For information about this service,press
```

```
1. "
    Text: "To start your order, press 2."
Input
    Max Number Of Digits: 01
Checklist
    Case: "1"
        Goto SPEAK_INFO
    Case: "2"
        Goto TAKE_ORDER
    Case: "Not On List"
        Reprompt
    Case: "Initial Timeout"
        Reprompt
    Case: "Too Few Digits"
        Reprompt
    Case: "No More Tries"
        Quit
End Prompt & Collect
SPEAK_INFO:
5. External Function
    Function Name: tts_file
    Use Arguments: 1  "/att/trans/sb/shopper/text"
6. Prompt & Collect
    Prompt
        Speak With Interrupt
        Text: "To start your order, press 1."
        Text: "To complete this call, press 2."
    Input
        Max Number Of Digits: 01
    Checklist
        Case: "1"
            Goto INITIATE_ORDER
        Case: "2"
            Goto GOODBYE
        Case: "Not On List"
            Reprompt
        Case: "Initial Timeout"
            Reprompt
        Case: "Too Few Digits"
            Reprompt
        Case: "No More Tries"
            Quit
    End Prompt & Collect

INITIATE_ORDER:
7. Prompt & Collect
    Prompt
        Speak With Interrupt
        Text: "Using touch tones, please enter your
```

```
5-digit"
    Text: "customer account number."
Input
    Min Number Of Digits: 05
    Max Number Of Digits: 05
Checklist
    Case: "Input Ok"
        Continue
    Case: "Initial Timeout"
        Reprompt
    Case: "Too Few Digits"
        Reprompt
    Case: "No More Tries"
        Quit
End Prompt & Collect
8. Read Table
    Table Name: cust_db      Search From Beginning
    Field: account_number = $CI_VALUE
9. Evaluate
    If $MATCH_FOUND != 0
10. Prompt & Collect
    Prompt
        Speak With Interrupt
        Field: NAME_RISKY_MODE As A
        Text: "Your name is: "
        Field: customer_name As A
        Text: "Press 1 for yes, 2 for no."
        Field: NAME_DEFAULT_MODE As A
    Input
        Max Number Of Digits: 01
    Checklist
        Case: "1"
            Goto CHECK_ADDRESS
        Case: "2"
            Goto TRANSFER_CALL
        Case: "Not On List"
            Reprompt
        Case: "Initial Timeout"
            Reprompt
        Case: "Too Few Digits"
            Reprompt
        Case: "No More Tries"
            Quit
    End Prompt & Collect
Else
11. Goto INITIATE_ORDER
End Evaluate

CHECK_ADDRESS:
### Verify the caller's address
12. Prompt & Collect
    Prompt
        Speak With Interrupt
        Field: ADDRESS_RISKY_MODE As A
```

```
Text: "Your address is "  
Field: customer_address As A  
Phrase: "sil.500"  
Field: customer_city As A  
Text: "Press 1 for yes, 2 for no."  
Input  
Max Number Of Digits: 01  
Checklist  
Case: "1"  
Goto TAKE_ORDER  
Case: "2"  
Goto TRANSFER_CALL  
Case: "Not On List"  
Reprompt  
Case: "Initial Timeout"  
Reprompt  
Case: "Too Few Digits"  
Reprompt  
Case: "No More Tries"  
Quit  
End Prompt & Collect  
  
TAKE_ORDER:  
13. Prompt & Collect  
Prompt  
Speak With Interrupt  
Text: "Enter the item number, or the pound
```

```
key to "  
    Text: "complete your order."  
Input  
    Max Number Of Digits: 05  
Checklist  
    Case: "#"  
        Goto WRAPUP_ORDER  
    Case: "nnnnn"  
        Continue  
    Case: "Not On List"  
        Reprompt  
    Case: "Initial Timeout"  
        Reprompt  
    Case: "Too Few Digits"  
        Reprompt  
    Case: "No More Tries"  
        Quit  
End Prompt & Collect  
14. Read Table  
    Table Name: grocery_db      Search From Beginning  
    Field: item_number = $CI_VALUE  
15. Evaluate  
    If $MATCH_FOUND != 0  
        ### Speak the item description and cost  
16. Announce  
    Speak With Interrupt  
    Field: item_description As A  
    Text: "for"  
    Field: item_price As AN$D2  
    ### Add the item to the customer's order  
17. Modify Table  
    Table Name: order_db Operation: Add  
    Field: order_number = account_number  
    Field: order_item = item_number  
18. Goto TAKE_ORDER  
End Evaluate  
  
WRAPUP_ORDER:  
19. External Function  
    Text: "Your order number is: "  
    Field: SPEAKING_RATE_FOUR as A  
    Field: order_number as A  
    Text: " . "  
    Text: "Please use this number when checking your
```

order."

GOODBYE:

20. Announce  
    Speak With Interrupt  
        Text: "Thanks for using the Home Shopping Line."  
        Text: "For further inquiries, please call"  
        Text: " npr 1 800-SHOPPER. npc"  
        Text: "Goodbye."

21. Disconnect

22. Quit

TRANSFER\_CALL:

23. Announce  
    Speak With Interrupt  
        Text: "Your call will be transferred to the next"  
        Text: "available agent."

24. Transfer To AGENT\_EXT Type: Intelligent  
    Maximum Number of Rings: 3  
    Case: "Answer"  
        Complete  
    Case: "Busy"  
        Reconnect  
    Case: "No Answer"  
        Reconnect  
    Case: "Error"  
        Reconnect  
    End Transfer

25. Disconnect

26. Quit

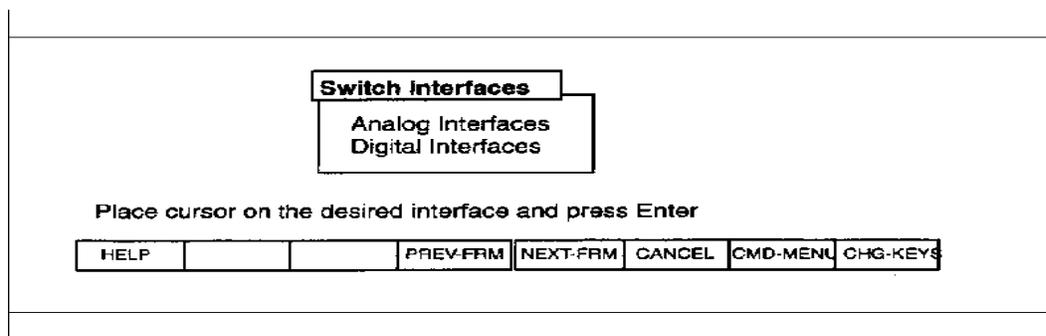
## Changing the Volume Level

---

To change the volume level of Text-To-Speech, perform the following procedure:

1. Enter **cvis\_menu**
2. Select Voice System Administration from the CONVERSANT VIS Version 5.0 screen.
3. Select Switch Interfaces to open the Switch Interfaces window (Figure D-1) from the Voice System Administration screen. The VIS must be running before you can use Switch Interface Administration. (To start the VIS, either use the System Control screen, or from the UNIX system command line, use the `start_vs` command). Refer to the information under "Start Voice System" in Chapter 3, "Configuration Management" in *Intuity CONVERSANT Voice Information System Version 5.0 Operations*, 585-310-550 for details on how to start the VIS.

The Switch Interfaces window offers two choices: Analog Interfaces and Digital Interfaces.



---

Figure D-1. Switch Interfaces Window

4. Highlight Analog Interfaces or Digital Interfaces (whichever applies to your platform) to open the screen. Figure D-2 demonstrates the Analog Interfaces screen.
5. Adjust the volume control. In the Outgoing Text Volume field, you can adjust the number to reflect the desired volume control.
6. Press **SAVE** to save your changes.
7. Press **CANCEL** to exit the window.

<b>Analog Interfaces</b>	
Switch Hook Flash Duration:	_____
Wink Disconnect Interval:	_____
Type of Signaling:	_____
Incoming Speech Volume:	_____
Outgoing Speech Volume:	_____
Outgoing Text Volume:	_____
Dial-Tone Training:	_____
Blind Transfer Actions	
To Initiate Transfer:	_____
To Complete Transfer:	_____
Intelligent Transfer Actions	
To Initiate Transfer:	_____
To Complete Transfer:	_____
To Reconnect Caller	
No Answer:	_____
Busy:	_____

HELP	CHOICES	SAVE	PREV-FRM	NEXT-FRM	CANCEL	CMD-MENU	CHG-KEYS
------	---------	------	----------	----------	--------	----------	----------

**Figure D-2. Analog Interfaces Window**

**Relating the Incoming Text Volume to Outgoing Text Volume**

This section includes an equation for calculating either the incoming text volume (IVOL), outgoing text volume (OVOL), or either (XVOL).

$$\text{Power dB} = 10 \log \frac{P_2}{P_1}$$
$$\frac{P_2}{P_1} = \left( \frac{\text{XVOL}}{1000} \right)^2$$

$$\text{Power dB} = 10 \log \left( \frac{\text{XVOL}}{1000} \right)^2$$

Let's say that the OVOL is set to 2000, how much dB gain is that?

$$10 \log (2)^2 = 10 \log (4) = + 6\text{dB gain}$$

What should IVOL be set to for a 3dB volume reduction?

$$\log \left( \frac{\text{IVOL}}{1000} \right)^2 = \frac{-3}{10}$$

$$\text{IVOL} = 707$$

---

# Abbreviations

---

## A

### AC

Alternating current

### ACD

Automatic call distributor

### AD

Application Dispatch

### AD-API

Application dispatch application programming interface

### ADPCM

Adaptive differential pulse code modulation

### ADU

Asynchronous data unit

### AGL

Application generation language

### ALERT

VIS Alerter process

### ANI

Automatic number identification

### API

Application programming interface

### ARU

Alarm relay unit

### ASAI

Adjunct/Switch Application Interface

### ASCII

American Standard Code for Information Interchange

### ASI

Analog switch integration

---

## B

### BB

Bulletin board

## Abbreviations

---

**bps**

Bits per second

**BRDG**

Call bridging process

**BSC**

Binary synchronous communication

---

**C****CCA**

Call classification analysis

**CDH**

Call data handler

**CELP**

Continuously Excited Linear Prediction

**CGEN**

Voice system general message class

**CICS**

Customer Information Control System

**CMP**

Companion circuit card

**CMS**

Call Management System

**CO**

Central office

**CPE**

Customer provided equipment or customer premise equipment

**CPN**

Calling party number

**CPT**

Call progress tones

**CPU**

Central processing unit

**CSU**

Channel service unit

**CVS**

Converse vector step

---

**D**

**dB**

Decibels

**DB**

Database

**DBC**

Database checking process

**DBMS**

Database management system

**DC**

Direct current

**DCE**

Data communications equipment

**DCP**

Digital communications protocol

**DIO**

Disk input and output process

**DIP**

Data interface process

**DMA**

Direct memory access

**DNIS**

Dialed number identification service

**DSP**

Digital signal processor

**DTE**

Data terminal equipment

**DTMF**

Dual tone multi-frequency

**DTR**

Data terminal ready

---

**E**

**EBCDIC**

Extended Binary Coded Decimal Interexchange Code

**EIA**

Electronic Industries Association

## Abbreviations

---

### **EISA**

Extended Industry Standard Architecture

### **EMI**

Electromagnetic interference

### **ESD**

Electrostatic discharge

### **ESDI**

Extended Serial Data Interface

### **ESS**

Electronic Switching System

### **ET**

Error tracker

### **EXTA**

External alarms feature message class

---

## **F**

### **FCC**

Federal Communications Commission

### **FDD**

Floppy disk drive

### **FEP**

Front end processor

### **FFE**

Form Filler Plus feature message class

### **FIFO**

First-in-first-out processing order

### **foos**

Facility out-of-service state

### **FTS**

File transfer process message class

---

## **G**

### **GEN**

PRISM logger and alerter general message class

### **GSE**

Graphical Speech Editor

### **GUI**

Graphical user interface

---

## **H**

### **HDD**

Hard disk drive

### **HLLAPI**

High Level Language Application Programming Interface

### **HOST**

Host interface process message class

### **hwoos**

Hardware out-of-service state

### **Hz**

Hertz

---

## **I**

### **IBM**

International Business Machines

### **ICK**

Integrity checking process message class

### **ID**

Identification

### **IDE**

Integrated Disk Electronics

### **IE**

Information element

### **INIT**

Voice system initialization message class

### **inserv**

In-service state

### **IPC**

Interprocess communication

### **IPC**

Intelligent Ports Card (IPC-900)

### **IPCI**

Integrated personal computer interface

### **IRAPI**

Intuity Response Application Programming Interface

### **IRQ**

Interrupt request

## Abbreviations

---

### **ISA**

Industry Standard Architecture

### **ISDN**

Integrated Services Digital Network

### **ISV**

Independent Software Vendor

### **ITAC**

International Technical Assistance Center

### **IVP4**

Integrated Voice Processing card with 4 analog channels

### **IVP6**

Integrated Voice Processing card with 6 analog channels

### **IVPSS**

Integrated Voice Processing System Software

---

## **K**

### **Kbps**

Kilobites per second

### **Kbyte**

Kilobyte

---

## **L**

### **LAN**

Local area network

### **LDB**

Local database

### **LED**

Light-emitting diode

### **LIFO**

Last-in-first-out processing order

### **LN**

Load number

### **LOG**

VIS logger process message class

### **LST1**

Line side T1

### **LU**

Logical unit

---

## M

**manoos**

Manually out-of-service state

**MAP/100**

Multi-Application Platform 100

**MAP/100C**

Multi-Application Platform 100C

**MAP/40**

Multi-Application Platform 40

**Mbps**

Megabits per second

**Mbyte**

Megabyte

**ms**

Millisecond

**msec**

Millisecond

**MHz**

Megahertz

**MTC**

Maintenance process

---

## N

**NCP**

Network Control Program

**NEBS**

Network Equipment Building Standards

**NEMA**

National Electrical Manufacturers Association

**netoos**

Network out-of-service state

**NFAS**

Non-Facility Associated Signaling

**NFS**

Network file sharing

**NMVT**

Network Management Vector Transport

## Abbreviations

---

### **NM-API**

Network Management - Application Programming Interface

### **nonex**

Nonexistent state

### **NRZ**

Non Return to Zero

### **NRZI**

Non Return to Zero Inverted

---

## **O**

### **OEM**

Original equipment manufacturer

### **OGA**

Operator generated alert

---

## **P**

### **PBX**

Private branch exchange

### **PC**

Personal computer

### **PCB**

Printed circuit board

### **PCM**

Pulse code modulation

### **PEC**

Price element code

### **PRI**

Primary rate interface

### **PSTN**

Public switch telephone network

### **PS&BM**

Power supply and battery module

---

## **R**

### **RAM**

Random access memory

## Abbreviations

---

**RECOG**

Speech recognition feature message class

**RDBMS**

ORACLE relational database management system

**REN**

Ringer equivalence number

**RFS**

Remote file sharing

**RM**

Resource manager

**RMB**

Remote maintenance board

**RTS**

Request to send

---

## S

**SBC**

Sub-band coding

**SCCS**

Switching Control Center System

**SCSI**

Small Computer System Interface

**SDLC**

Synchronous Data Link Control

**SDN**

Software Defined Network

**SID**

Station identification

**SIMM**

Single inline memory module

**SLIP**

Serial Line Interface Protocol

**SNA**

Systems Network Architecture

**SNMP**

Simple Network Management Protocol

**SP**

Signal processor circuit card

## Abbreviations

---

**SPIP**

Signal processor interface process

**SPPLIB**

Speech processing library

**SQL**

Structured Query Language

**SR**

Speech recognition

**SYS**

UNIX system calls message class

**sysgen**

System generation

---

## T

**tas**

Transaction assembler

**TCC**

Technology Control Center

**TCP/IP**

Transmission control protocol/internet protocol

**TDM**

Time division multiplexing

**TE**

Terminal emulator

**THR**

Threshold message class

**TKR**

Token Ring

**TLI**

Transport layer interface

**TLP**

Transmission level plan

**T/R**

Tip/Ring circuit card

**TRIP**

Tip/Ring interface process

**TSO**

Technical Service Organization

## Abbreviations

---

### **TSO**

Time Share Operation

### **TSM**

Transaction state machine process

### **TTS**

Text-to-Speech

### **TWIP**

T1 interface process

---

## **U**

### **UK**

United Kingdom

### **USOC**

Universal service ordering code

### **UVL**

Unified Voice Library

---

## **V**

### **VDC**

Video display controller

### **VIS**

Intuity CONVERSANT Voice Information System

### **VPC**

Voice processing comarketer

### **VRU**

Voice response unit

### **VROP**

Voice response output process



---

# Glossary

---

## Numerics

---

### **3270 interface**

A link between one or more Intuity CONVERSANT Voice Information System (VIS) machines and a host mainframe. In Intuity CONVERSANT VIS documentation, the 3270 interface means the link between one or more VIS machines and an IBM host mainframe.

### **4ESS**

A large AT&T central office switch used to route calls through AT&T's telephone network.

---

## A

### **ACD**

See "automatic call distributor."

### **ADPCM**

See "adaptive differential pulse code modulation."

### **adaptive differential pulse code modulation**

A means of encoding analog voice signals into digital signals by adaptively predicting future encoded voice signals. This adaptive modulation method reduces the number of bits required to encode voice. See also "pulse code modulation."

### **adjunct products**

Products (for example, Adjunct/Switch Application Interface) that the Intuity VIS administers via cut-through access to the inherent management capabilities of the product itself; this is in opposition to CONVERSANT VIS's ability to administer the switch directly.

### **Adjunct/Switch Application Interface**

An optional feature package that provides an Integrated Services Digital Network-based interface between AT&T PBX's and adjunct processors.

### **affiliate**

A business organization that AT&T controls or which with AT&T is in partnership.

### **alarm relay unit**

A unit used in central office telecommunication arrangements that transmits warning indicators from telephone communications equipment (like the Intuity CONVERSANT VIS) to audio.

### **alerter**

A system process that responds to patterns of events logged by the "logdaemon" process.

**analog**

An analog signal, such as voice or music, that varies in a continuous manner. An analog signal may be contrasted with a digital signal, which represents only discrete states.

**application**

Made of several components that provide an automated version of the communication between a caller and an attendant. The Intuity CONVERSANT VIS provides several methods for creating applications, including Script Builder, the Intuity Response Application Programming Interface (IRAPI), and transaction state machine (TSM) script language.

**application administration**

The component of the Intuity CONVERSANT VIS that provides access to the applications currently available on your system and helps you to manage and administer them.

**application installation**

A two-step process in which the Intuity CONVERSANT VIS invokes the TSM script assembler for the specific application name and files are moved to the appropriate directories.

**application verification**

A process in which the Intuity CONVERSANT VIS verifies that all the components needed by an application are complete.

**ASCII**

An acronym for American Standard Code for Information Interchange, a standard for data representation. ASCII code represents alphanumeric characters as binary numbers. The code includes 128 upper- and lowercase letters, numerals, and special characters. Each alphanumeric and special character has an ASCII code (binary) equivalent that is 1 byte long.

**asynchronous communication**

A method of data transmission in which bits or characters are sent at irregular intervals and are spaced by start and stop bits and not by time. See also "synchronous communication."

**asynchronous data unit**

An electronic communications device that allows computer systems to communicate over asynchronous lines more than 50 feet in length.

**AUDIX Voice Power**

A complete voice-mail messaging system accessed and operated by touch-tone telephones and integrated with a switch or "Private Branch Exchange."

**automatic call distributor**

A telephone system that recognizes and answers incoming calls and completes these calls based on a set of instructions contained in a database. The Automatic Call Distributor can send the call to an operator or group of operators as soon as the operator has completed a previous call or after the system has played a message to the caller.

**automatic number identification**

A method of identifying the calling party by automatically receiving a string of digits that identifies the calling station of a particular customer.

---

## B

### **back up**

The preservation of the information in a file in a different location, so that the data is not lost in the event of hardware or system failure.

### **backing up an application**

A utility that makes an archive copy of a completed application or makes an interim copy of an application in progress. The backup copy can be restored to the VIS if the online version is damaged, or if you make revisions and wish to go back to the previous version.

### **barge-in**

A capability provided by WholeWord speech recognition that allow callers to speak their responses to the VIS prompt and have those responses recognized before the prompt has finished playing.

### **batch file**

A file containing one or more lines, each of which is a command executable by the UNIX shell.

### **binary synchronous communications**

A character-oriented synchronous link protocol.

### **blind transfer protocol**

A protocol in which a call is completed as soon as the extension is dialed, without having to wait to see if the telephone is busy or if the caller answered.

### **bridging**

The process of connecting one telephone network connection to another telephone network connection over the Intuity CONVERSANT VIS TDM bus. Bridging decreases the processing load on the system since an active bridge does not require speech processing, database access, host activity, etc., for the transaction.

### **BSC**

See "binary synchronous communication."

### **bundle**

In the context of the Enhanced File Transfer package, this term is used to denote a single file, a group of files (package), or a combination of both.

### **byte**

A unit of storage in the computer. On many systems, a byte is 8 bits (binary digits), the equivalent of one character of text.

---

## C

### **call classification analysis**

An optional feature package that allows application developers to classify the disposition of originated and transferred calls.

**call data event**

A parameter that specifies a list of variables that are appended to a call data record at the end of each call.

**call data handler process**

A software process that accumulates generic call statistics and application events.

**called party number**

The number dialed by someone making a telephone call. It can be used by telephone switching equipment to selectively route an incoming call to a particular department or agent.

**caller**

The party that calls for a service, gets connected to the Intuity CONVERSANT VIS, and interacts with the system. As the Intuity CONVERSANT VIS is also capable of making outbound calls for service, the caller can also be the person who responds to those outbound calls.

**call progress tones**

Standard telephony sounds that indicate the status of the call. These sounds include busy, fast busy, ringback, reorder, etc.

**card cage**

An area within a Intuity CONVERSANT VIS platform that contains and secures all of the standard and optional circuit cards used in the system.

**cartridge tape drive**

A high-capacity data storage/retrieval device that can be used to transfer large amounts of information onto high-density magnetic cartridge tape based on a predetermined format. This tape can be removed from the system and stored as a backup, or used on another system.

**caution**

An admonishment used when there is a possibility of a service interruption or a loss of data.

**CCA**

See "call classification analysis."

**CDH**

See "call data handler process."

**central office**

An office or location in which large telecommunication machines such as telephone switches and network access facilities are maintained. These locations follow strict installation and operation requirements.

**central processing unit**

A component of the Intuity CONVERSANT VIS that is based on either the Multi-Application Platform 100 (MAP/100), MAP/40, or MAP/100C.

**channel**

See "port."

**CICS**

See "Customer Information Control System."

**circuit card upgrade**

A new circuit card that replaces an existing one in the platform. Usually the replacement is an updated version of the other card, and the replacement is designed to deal with technology made obsolete by industry trends or a new VIS release.

**cluster controller**

A bisynchronous interface that provides a means of handling remote communication processing.

**command**

An instruction or request given by the user to the VIS software to perform a particular function. An entire command consists of the command name and options.

**CompuLert/SCCS interface**

An optional feature that enables remote or console monitoring of error messages generated from the Intuity CONVERSANT VIS. CompuLert is a centralized maintenance system for monitoring minicomputers, computer mainframes, etc. The Switching Control Center System (SCCS) is similar to the CompuLert system, but is used to support 4ESS local switching systems.

**configuration**

The arrangement of the software and hardware of a computer system or network. The Intuity CONVERSANT VIS configuration includes either a standard or custom processor, peripheral equipment (for example, printers, modems), and software applications. Configuration also refers to the way the switch network is set up; that is, the types of products that are in the network and how those products communicate.

**configuration management**

The component of the VIS that allows you to manage the current configuration of voice channels, host sessions, and database connections, assign scripts to run on specific voice channels or host sessions assign functionality to SP and T1 cards, and perform various maintenance functions.

**Converse Data Return (conv\_data)**

A Script Builder action that supports the DEFINITY call vectoring (routing) feature by enabling the switch to retain control of vector processing in the VIS environment. It supports the DEFINITY "converse" vector command to establish a two-way routing mechanism between the switch and the VIS to facilitate data passing and return.

**controller circuit card**

A circuit card used on a computer system that controls its basic functionality and makes the system operational. These cards are used to control magnetic peripherals, video monitors, and basic system communications.

**copying an application**

A utility in which information from a source application is directed into the destination application.

**coresidency**

The ability of two products or services to operate and interact with each other on a single hardware platform. An example of this is the use of AUDIX Voice Power along with Intuity CONVERSANT on the same VIS platform.

**CPU**

See "central processing unit."

**crash**

An interactive utility for examining the operating system core and for determining if system parameters are being exceeded.

**custom speech**

Unique words or phrases to be used in Intuity CONVERSANT VIS voice prompts that AT&T records for a customer on a custom basis.

**custom vocabulary**

A specialized package of unique words or phrased created on a per-customer basis and used by WholeWord or FlexWord speech recognition.

**Customer Information Control System**

Part of the operating system that manages resources for running applications (for example, IND\$FILE). Note that TSO and CMS provide analogous functionality in other host environments.

---

**D**

**danger**

An admonishment used when there is a possibility of personal injury.

**data interface process**

A software process that communicates with Script Builder applications.

**database**

A structured set of files, records, or tables.

**database field**

A field used to extract values from a local database and form the structure upon which a database is built.

**database table**

A structure, made up of columns and rows, that holds information in a database. Database tables provide a means of storing information that changes too often to “hard-code,” or permanently store, in the transaction outline.

**debug**

The process of locating and correcting errors in computer programs. This process is also referred to as “troubleshooting.”

**default**

The way a computer performs a task in the absence of other instructions.

**default owner**

The owner of a channel when no process takes ownership of that channel. The default owner holds all idle, in-service channels. In terms of the IRAPI, this is typically the Application Dispatch process.

**diagnose**

The process of performing diagnostics on Tip/Ring, T1, or SP circuit cards or a bus.

**dialed number identification service**

A service that allows incoming calls to contain information about the telephone number for which it is destined.

**directory**

A type of file used to group and organize other files or directories.

**DNIS**

See “dialed number identification service.”

**DIP**

See “data interface process.”

**display errdata**

A command that displays system errors sent to the logger.

**DTMF**

See "dual tone multi-frequency."

**dual 3270 links**

A feature that provides an additional physical unit (PU) to allow a cost-effective means of connecting to two host computers. The customer can connect a VIS to two separate FEPs or to a single FEP shared by one or more host computers. Each link supports a maximum of 32 LUs.

**dual tone multi-frequency**

A touch tone.

**dump space**

An area of the disk that is fixed in size and should equal the amount of RAM on the system. The operating system "dumps" an image of core memory upon system crashes. The dump can be fetched after rebooting for analysis of what may have caused the crash.

---

**E**

**editor system**

A system that allows speech phrases to be displayed and edited by a user. See "Graphical Speech Editor."

**Enhanced File Transfer**

A feature that allows the transferring of files automatically between the Intuity CONVERSANT VIS and a synchronous host processor on a designated logical unit.

**Enhanced Serial Data Interface**

A software- and hardware-controlled method used to store data on magnetic peripherals.

**error message**

A message on the screen indicating that something is wrong and possibly suggesting how to correct it.

**Error Tracker process**

See "etStub."

**Ethernet**

A name for a local area network that uses 10BASE5 or 10BASE2 coaxial cable and InterLAN signaling techniques.

**etStub**

A system process that processes pre-Version 3.1 error message logging requests. These requests are transformed and passed on to the "logdaemon" process.

**event**

The notification given to an application when some condition occurs.

**external actions**

Specific tasks and interfaces controlled by Intuity CONVERSANT VIS software that allow a Script Builder application script to invoke processes and interact with other products or services. For example, a Intuity CONVERSANT VIS application script can invoke AUDIX Voice Power functionality through the used of an external action within an application script.

---

**F**

**feature**

A function or capability of a product or an application within the Intuity CONVERSANT VIS.

**feature package**

An optionally purchased package that may contain both hardware and software resources, which provides additional functionality to a standard system.

**feature\_tst script package**

A standard CONVERSANT VIS software program that allows a VIS user to perform self-tests of critical hardware and software functionality.

**field**

A "slot" in a VIS window that holds one column of information in a row.

**file**

A collection of data treated as a basic unit of storage.

**file transfer**

An option that allows you to transfer files interactively or directly to and from UNIX using the File Transfer System.

**filename**

Alphabetic characters used to identify a particular file.

**FlexWord speech recognition**

A type of speech recognition based on subword technology that recognizes phonemes or parts of words of American English vocabularies. See "subword technology."

**Form Filler Plus**

An optional feature package that provides the capability for application scripts to record caller's responses to prompts for later transcription and review.

**function key**

A key, labeled F1 through F8, on your keyboard to which the Intuity CONVERSANT VIS software gives special properties for manipulating the user interface.

---

## G

### **Graphical Speech Editor**

A window-driven, X Windows/Motif based, graphical user interface (GUI) that can be accessed to perform different functions associated with the creation and editing of speech files to be used by VIS applications.

---

## H

### **hard disk drive**

A high-capacity data storage/retrieval device that is located inside a computer platform. A hard disk drive stores data on nonremovable high-density magnetic media based on a predetermined format for retrieval by the system at a later date.

### **hardware**

The physical components of a computer system. The central processing unit, disks, tape and floppy drives, etc., are all hardware.

### **hardware upgrade**

Replacement of one or more fundamental platform hardware components (for example, the CPU or hard disk drive), but the existing platform and other existing optional circuit cards remain.

### **High Level Language Applications Programming Interface (HLLAPI)**

An application programming interface that allows user to write custom applications that can communicate with the host via an API.

### **HLLAPI**

See "High Level Language Applications Programming Interface."

### **host computer**

A computer linked to a network providing a range of services, such as database access and computation. The host computer operates in a time-sharing manner with other computers linked to it via the network.

---

## I

### **iCk**

The system integrity checking process.

### **idle channel**

A channel that either has no owner or is owned by its default owner and is onhook.

### **IND\$FILE**

The standard SNA file transfer utility that runs as an application under CICS, TSO, and CMS. IND\$FILE is independent of link-level protocols such as BISYNC and SDLC.

### **indexed table**

A table that, unlike a nonindexed table, can be searched via a field name that has been indexed.

**initialize**

To start up the system for the first time.

**Integrated Services Digital Network**

A network that provides end-to-end digital connectivity to support a wide range of voice and data services.

**Integrated Voice Processing circuit card**

The IVP4 or IVP6 circuit card.

**intelligent transfer protocol**

A transfer protocol that monitors the line after dialing is complete to determine whether a busy, reorder (fast busy), or other failure has been encountered. It also recognizes when the extension is answered or if the extension is not answered after a specified number of rings.

**interface**

The access point of a system. With respect to the Intuity CONVERSANT VIS, the interface is designed to provide you with easy access to the software's capabilities.

**interrupt**

The termination of voice and/or telephony functions when some condition occurs.

**Intuity Response Application Programming Interface**

A library interface that provides a standard development interface for voice-telephony applications.

**ipcs**

A command that reports interprocess communication facilities status.

**IRAPI**

See "Intuity Response Application Programming Interface."

**ISDN**

See "Integrated Services Digital Network."

---

**K**

**keyboard mapping**

In emulation mode, this feature enables the keyboard to send 3270 keyboard codes to the host according to a configuration table set up during installation.

**keyword spotting**

A capability provided by WholeWord Speech Recognition that allows the VIS to recognize a single word in the middle of an entire phrase spoken by a caller in response to a prompt.

---

**L**

**LAN**

See "local area network."

**library states**

The state information about channel activities maintained by the IRAPI.

**line side T1**

A digital method of interfacing a Intuity CONVERSANT VIS to a PBX or switch using T1-related hardware and software.

**listfile**

An ASCII catalog that lists the contents of one or more talkfiles. Each application script is typically associated with a separate listfile. The listfile maps speech phrase strings used by application scripts into speech phrase numbers.

**local area network**

A data communications network in a limited geographical area. The local area network provides communications between computers and peripherals.

**local database**

A database residing on the Intuity CONVERSANT VIS.

**logical unit**

A type of SNA Network Addressable Unit.

**logdaemon**

System information and error logging process.

**logger**

See "logdaemon."

**logging on/off**

Entering or exiting the Intuity CONVERSANT VIS software.

**LU**

See "logical unit."

---

## M

**magnetic peripherals**

Data storage devices that use magnetic media to store information. Such devices include hard disk drives, floppy disk drives, and cartridge tape drives.

**main screen**

The Intuity CONVERSANT VIS VERSION 5.0 screen from which you are able to enter System Administration or Voice System Administration.

**maintenance process**

A software process that runs temporary diagnostics.

**Manual Configurator Program**

A software program that resolves or blocks the allocation of CPU and memory resources for controlling and optional circuit cards.

**masked event**

An event that an application can ignore (that is, the application can ask not to be informed of the event).

**master**

A board that provides clock information to the TDM bus.

**megabyte**

A unit of memory equal to 1,048,576 bytes (1024 x 1024). It is often rounded to one million.

**Microsoft**

A company that manufactures software products, primarily for IBM-compatible computers.

**mirroring**

A method of data backup that allows all of the data transactions to the primary hard disk drive to be copied and maintained on a second identical drive in near real time. If the primary disk drive crashes or becomes disabled, all of the data stored on it (up to 1.2 billion bytes of information) is accessible on the second mirrored disk drive.

**MS-DOS**

A personal computer disk operating system developed by the Microsoft Corporation.

**MTC**

See "maintenance process."

**multi-threaded application**

A single process/application that controls several channels. Each thread of the application is managed explicitly. Typically this means state information for each thread is maintained and the state of the application on each channel is tracked.

---

**N**

**NetView**

An optional feature package that transmits high-priority (major or critical) messages to the host as Operator-Generated Alerts (OGAs) over the 3270 host link. The NetView Alarm feature package does not require a dedicated LU.

**new error logging environment**

A more flexible and informative environment for logging errors and status messages (introduced in CONVERSANT VIS Version 3.1). Customer applications created earlier than V3.1 that log messages require conversion to this new environment.

**new operating system**

The UnixWare operating system being introduced in Intuity CONVERSANT VIS V5.0.

**nonindexed table**

A table that may be searched only in a sequential manner and that cannot be searched via a field name.

**nonmasked event**

An event that must be sent to the application. Generally, an event is nonmaskable if the applicaiton would likely encountered state transition errors by trying to ignore the event.

**null value**

An entry containing no value. A field containing a null value is normally displayed as blank and is different from a field containing a value of zero.

---

## O

### **obsolete hardware**

Hardware that is no longer supported on Intuity CONVERSANT VIS V5.0.

### **on-line help**

Messages or information that appear on the user's screen when a "function key" (F1 through F8) is pressed.

### **Operator Generated Alerts**

System monitoring messages transmitted from the CONVERSANT VIS or other computer system to an IBM host computer that are classified as critical or major.

### **option**

An argument used in a command line to modify program output by modifying the execution of a command. When you do not specify any options, the command will execute according to its default options.

### **ORACLE**

A company that produces Relational Database Management software. It is also used as a generic term that identifies a database residing on a local or remote system that is created and maintained using an ORACLE RDBMS product.

---

## P

### **PBX**

See "private branch exchange."

### **PCM**

See "pulse code modulation."

### **peripheral (device)**

Equipment such as printers or terminals that is in addition to the basic processor.

### **permanent process**

A process that starts and initializes itself before it is needed by a caller.

### **phoneme**

A single basic sound of particular spoken language. The English language contains 40 phonemes that represent all basic sounds used with the language. As an example, the word "one" can be represented with three phonemes, "w" - "uh" - "n." Phonemes vary between languages because of guttural and nasal inflections and syllable constructs.

### **phrase filtering**

The rejection of unrecognized speech. The WholeWord and FlexWord speech recognition packages can be programmed to reprompt the caller if the spoken response was not recognized by the VIS.

### **phrase tag**

A string of up to 50 characters that identify the contents of a speech phrase used by an application script.

**platform migration**

See "platform upgrade."

**platform upgrade**

The process of replacing the existing platform with a new platform.

**poll**

A message sent from a central controller to an individual station on a multipoint network inviting that station to send if it has any traffic to send.

**polling**

A network arrangement whereby a central computer asks each remote location whether they wish to send information. This arrangement enables each user or remote data terminal to transmit and receive information on shared facilities.

**port**

A connection or link between two devices that allows information to travel to a desired location. See "telephone network connection."

**Primary Rate Interface**

An optional feature package that provides a digital interface capable both of receiving and originating telephone calls directly from/to an AT&T 4ESS switch.

**private branch exchange**

A private switching system, either manual or automatic, usually serving an organization, such as a business or government agency, and usually located on the customer's premises.

**processor**

In Intuity CONVERSANT VIS documentation, the computer on which UnixWare and Intuity CONVERSANT VIS software runs. In general, the part of the computer system that processes the data. Also known as the "central processing unit."

**ps**

A command that shows active processes. This command displays the process table and can be used to determine which processes are consuming large amounts of system resources, such as CPU time.

**pseudo driver**

A driver that does not control any hardware.

**pulse code modulation**

A digital modulation method of encoding voice signals into digital signals. See also "adaptive differential pulse code modulation."

---

## R

**recovery**

The process of using copies of the VIS software to reconstruct files that have been lost or damaged. See also "restore."

**remote database**

The component of the VIS that provides access to information not currently on the VIS.

**remote maintenance board**

A Intuity CONVERSANT VIS board that is equipped standard on all new MAP/100 and MAP/40 platform purchases. This card, available with a built-in modem, allows remote personnel (for example, field support) to access all Intuity CONVERSANT VIS machines with a standard simplified process.

**reports administration**

The component of the VIS that provides access to system reports, including VIS call classification reports, call data detail reports, call data summary reports, message log reports, and traffic reports. In addition, if AUDIX Voice Power R2.1.1 is installed on your system, the reports administration component gives you access to AUDIX Voice Power reports.

**restore**

The process of recovering lost or damaged files by retrieving them from available backup tapes or from another disk device. See also "recovery."

**restore application**

A utility that replaces a damaged application or restores an older version of an application.

**reuse**

The concept of reusing an existing system component after a software upgrade or platform migration.

**roll back**

To cancel changes to a database since the point at which changes were last committed.

**rollback segment**

A portion of the database that records actions that should be undone under certain circumstances. Rollback segments are used to provide transaction rollback, read consistency, and recovery.

---

**S**

**sar**

A command that is associated with the system activity report package.

**screen pop**

A method of delivering a screen of information to a telephone operator at the same time a telephone call is delivered. This is accomplished by a complex chain of tasks that include identifying the calling party number, using that information to access a local or remote ORACLE database, and pulling a "form" full of information from the database using an ORACLE database utility package.

**script**

The set of instructions for the Intuity CONVERSANT VIS to follow during a transaction.

**Script Builder**

An optional software package that provides a menu-oriented interface designed to assist in the development of custom voice response applications on the VIS.

**SCSI**

See "Small Computer System Interface."

**shared database table**

A database table that is used in more than one application.

**shared speech**

Speech that is a part of more than one application.

**shared speech pools**

A parameter that allows the user of a voice application to share speech components with other applications.

**Single Inline Memory Modules**

A method of containing random access memory (RAM) chips on narrow circuit card strips that attach directly to sockets on the CPU circuit card. Multiple SIMMs are sometimes installed on a single CPU circuit card.

**single-threaded application**

An application that runs on a single voice channel.

**slave**

A circuit card that depends on the TDM bus for clock information.

**Small Computer System Interface**

A disk drive control technology in which a single SCSI adapter card plugged into a PC slot is capable of controlling as many as seven different hard disks, optical disks, tape drives, etc.

**software**

The set or sets of programs that instruct the computer hardware to perform a task or series of tasks — for example, UnixWare software and the Intuity CONVERSANT VIS Version 5.0 software.

**software upgrade**

The installation of a new version of software. The existing platform and circuit cards are kept.

**source system**

The system from which you are upgrading (that is, your system as it exists *before* you upgrade).

**speech energy**

The amount of energy in an audio signal. Literally translated, it is the output level of the sound in every phonetic utterance.

**speech envelope**

The linear representation of voltage on a line. It reflects the sound wave amplitude at different intervals of time. This envelope can be plotted on a graph to represent the oscillation of an audio signal between the positive and negative extremes.

**speech file**

A file containing an encoded speech phrase.

**speech filesystem**

A collection of several talkfiles. The filesystem is organized into 16-Kbyte blocks for efficient management and retrieval of talkfiles. The Intuity CONVERSANT VIS speech filesystem is not consistent with standard UNIX filesystems, and can not be referenced with standard UNIX commands such as **ls**, **cat**, etc.

**speech modeling**

Creating WholeWord speech recognition algorithms by collecting thousands of different speech samples of a single word and comparing them all to obtain a statistical average of the word. This average is then used by a WholeWord speech recognition program to recognize a single spoken word.

**speech phrase**

A continuous speech segment encoded into a digital string.

**speech space**

An area that contains all digitized speech used for playback in the applications loaded on the system.

**standard speech**

The speech package containing simple words and phrases produced by AT&T for use with an Intuity CONVERSANT VIS. This package includes digits, numbers, days of the week, and months, each spoken with initial, medial, and falling inflection. The speech is in digitized files stored on the hard disk to be used in the voice prompts played by the VIS.

**standard vocabulary**

A standard package of simple word speech models provided by AT&T and used for WholeWord speech recognition purposes. These phrases include the digits "zero" through "nine," "yes," "no," and "oh."

**string**

A contiguous sequence of characters treated as a unit. Strings are normally bounded by white spaces, tabs, or a character designated as a separator. A string value is a specified group of characters symbolized by a variable.

**Structured Query Language**

A standard data programming language used with data storage and data query applications.

**subword technology**

A method of speech recognition that recognizes phonemes or parts of words of American English vocabularies. See "whole-word technology."

**switch**

A software and hardware device that controls and directs voice and data traffic. A customer-based switch is known as a "private branch exchange."

**switch hook**

The device at the top of most telephones that is depressed when the handset is resting in the cradle (on hook). The device is raised when the handset is picked up (the telephone is off hook).

**switch hook flash**

A signaling technique in which the signal is originated by momentarily depressing the "switch hook."

**switch interface administration**

The component of the VIS that enables you to define the interaction between the VIS and switches by allowing you to establish and modify switch interface parameters and protocol options for both analog and digital interfaces.

**switch network**

Two or more interconnected switching systems.

**synchronous communication**

A method of data transmission in which bits or characters are sent at regular time intervals, rather than being spaced by start and stop bits. See also "asynchronous communication."

**System 75**

An advanced digital switch supporting up to 800 lines that provides voice and data communications for its users.

**System 85**

An advanced digital switch supporting up to 3000 lines that provides voice and data communications for its users.

**system administrator**

The person assigned the responsibility of monitoring all VIS software processing, performing daily system operations and preventive maintenance, and troubleshooting errors as required.

**system architecture**

The manner in which the Intuity CONVERSANT VIS software is structured.

**system message**

An event or alarm generated by either a VIS or end-user process.

**system monitor**

A component of the VIS in which tests are performed to verify that each incoming telephone line and its associated tip/ring or T1 card is functional. Through the "System Monitor" component, you are able to see displays of the Voice Channel and Host Session Monitors.

---

**T**

**T1**

A digital transmission link with a capacity of 1.544 Mbps.

**table**

A collection of records that are logically grouped together.

**talkfile**

An ASCII file that contains the speech phrase tags and phrase tag numbers for all the phrases of a specific application. The speech phrases are organized and stored in groups. Each talkfile can contain up to 65,535 phrases and the speech filesystem can contain multiple talkfiles.

**target system**

The system to which you are upgrading (that is, your system as you expect it to exist *after* you upgrade).

**TDM**

See "time-division multiplex."

**telephone network connection**

The point at which a telephone network connection terminates on an Intuity CONVERSANT VIS. Supported telephone connections are Tip/Ring and T1.

**Terminal Emulator**

Software that allows the VIS to temporarily transform itself into a "look alike" of an IBM 3270 terminal. In addition to providing full 3270 functionality, the Terminal Emulator enables you to transfer files to and from UNIX.

**Text-to-Speech**

An optional feature that allows an application to play speech directly from ASCII text by converting that text to synthesized speech. The text can be used for prompts or for text retrieved from a database or host, and can be spoken in an application with prerecorded speech. Text-to-Speech application development is supported through Script Builder.

**ThickNet**

A 10-millimeter (10BASE5) coaxial cable used to provide InterLAN communications.

**ThinNet**

A 5-millimeter (10BASE2) coaxial cable used to provide InterLAN communications.

**time-division multiplex**

A method of serving a number of simultaneous channels over a common transmission path by assigning the transmission path sequentially to the channels, with each assignment being for a discrete time interval.

**Tip/Ring**

A term used to denote analog telecommunications using four-wire media.

**Token/Ring**

A ring type of local area network that allows any station in the network to communicate with any other station.

**trace**

A command that can be used to monitor the execution of a script.

**traffic**

The flow of information or messages through a communications network for voice, data, or audio services.

**transaction**

Comprised of the exchanges between the caller and the voice system. A transaction can involve one or more telephone network connections and voice responses from the Intuity CONVERSANT VIS. It can also involve one or more of the VIS optional features, such as speech recognition, 3270 host interface, FAX response, etc.

**transaction state machine process**

A multi-channel IRAPI application that runs applications driven by script information.

**transient process**

A process that is created dynamically only when needed.

**troubleshoot**

The process of locating and correcting errors in computer programs. This process is also referred to as debugging.

**TSM**

See "transaction state machine process."

**TTS**

See "Text-to-Speech."

---

**U**

**UNIX Operating System**

A multiuser, multitasking computer operating system developed by the Bell Telephone Laboratories division of AT&T.

**UNIX shell**

The command language that provides a user interface to the UNIX operating system.

**upgrade image tape**

A tape, optionally provided to you by the Technical Service Organization, containing the new operating system and Intuity CONVERSANT VIS V5.0 base software in a standard configuration which is compatible with your target system.

**upgrade scenario**

The particular combination of current hardware, software, application and target hardware, software, applications, etc.

---

**V**

**vi editor**

A screen editor used by the Intuity CONVERSANT VIS to create and change electronic files.

**virtual channel**

A channel that is not associated with an interface to the telephone network (Tip/Ring, T1, or PRI). Virtual channels are intended to run "data only" applications which do not interact with callers but may interact with DIPs. Voice or network functions (for example, coding or playing speech, call answer, origination, or transfer) will not work on a virtual channel. Virtual channel applications may be initiated only by a "virtual seizure" request to TSM from a DIP.

**VIS**

See "Voice Information System."

**vocabulary**

A collection of words that a VIS is able to recognize using either WholeWord or FlexWord speech recognition.

**vocabulary activation**

The set of active vocabularies that define the words and wordlists known to the FlexWord recognizer.

**vocabulary loading**

The process of copying the vocabulary from the system where it was developed and adding it to the target system.

**voice channel**

A channel that is associated with an interface to the telephone network (Tip/Ring, T1, or PRI). Any Intuity CONVERSANT VIS application can run on a voice channel. Voice channel applications may be initiated by being assigned to particular voice channels or dialed numbers to handle incoming calls or by a "soft seizure" request to TSM from a data interface process (DIP) or the **soft\_srz** command.

**Voice Information System**

A computer connected to a telephone network that handles touch-tone input, voice response, and line transfer. The Voice Information System uses a screen-based, menu-driven user interface to interact with the system operator or administrator.

**voice processing co-marketer**

A company licensed to purchase voice processing equipment, such as the Intuity CONVERSANT VIS, to market and sell based on their own marketing strategies.

**voice response output process**

A software process that transfers digitized speech between system hardware (for example, Tip/Ring and SP cards) and data storage devices (that is, hard disk, etc.)

**Voice System Administration**

The means by which you are able to administer both voice- and nonvoice-related aspects of the system.

**VROP**

See "voice response output process."

---

**W**

**warning**

An admonishment used when there is a possibility of equipment damage.

**WholeWord speech recognition**

An optional feature based on whole-word technology that provides speaker independence, connected digit recognition, key word spotting, prompt interrupt, and DTMF support functionality. See "whole-word technology."

**whole-word technology**

The ability to recognize an entire word, not the phoneme or a part of a word. See "subword technology."

**wink signal**

An interruption of current to a busy lamp indicating that there is a line on hold.

**word**

A unique utterance understood by the recognizer.

**wordlist**

A set of words identified by a wordlist name. If the wordlist is part of an active vocabulary, the wordlist name appears as a recognition type in the Prompt & Collect mode field.

**word spotting**

The ability to search past extraneous speech during a recognition.



---

# Index

---

## A

- Accessing the Graphical Speech Editor, 3-3
- Accuracy
  - negative influences on FlexWord Speech Recognition, 4-48
  - positive influences on FlexWord Speech Recognition, 4-45
  - WholeWord Speech Recognition versus FlexWord Speech Recognition, 4-50
- Accuracy of FlexWord Speech Recognition, 4-45
- Activating a FlexWord wordlist, 4-43
- Adaptive Differential Pulse Code Modulation, 3-2, B-2
- addhdr command, 3-37
- Adding a second speech disk
  - reference for, 1-3
- Adding words and phrases to a FlexWord vocabulary, 4-39
- Administering and installing the mouse, 3-3, 4-33
- ADPCM-16, B-2
- ADPCM-32, B-2
- Alpha characters
  - recording, 2-16
- Application
  - definition of, xi
- Application using escape sequences
  - Text-to-Speech, D-8
- AT&T custom vocabulary service for FlexWord Speech Recognition, 4-24
- Audience of book, xii
- Audio Works Station, 3-33

---

## B

- Barge-In for WholeWord Speech Recognition, 4-9
- Bilingual speech recognition capabilities, 4-2
- Book
  - conventions used, xiv
  - how to make comments, xvii
  - how to use, xiii
  - intended audiences, xii
  - organization of, xii
  - purpose of, xi
  - related resources to, xvi

- Building a FlexWord vocabulary, 5-3
- 

## C

- C application
  - installing speech, 2-20
- Caller error
  - FlexWord Speech Recognition, 5-4
- Changing class detection
  - using escape sequences in Text-to-Speech, D-6
- Changing the size of the speech filesystem
  - reference for, 1-3
- Changing the volume level
  - Text-to-Speech application, D-14
- Channels
  - adding a second speech disk, 1-3
- Choosing a FlexWord vocabulary, 5-3
- Coding rates for speech phrases, 1-2
- Combining WholeWord Speech Recognition and touch tone, 5-2
- Computing channel numbers
  - FlexWord toolkit, C-2
  - Graphical Speech Editor, C-2
- Concepts about speech, 1-5
- Configuring hardware for the FlexWord toolkit, 4-31
- Connected digit recognition
  - WholeWord Speech Recognition accuracy, 4-3
- Connected digit recognition for WholeWord Speech Recognition, 4-11
- Considering the sound of a word in a FlexWord speech recognition application, 5-4
- Conventions used in book, xiv
- Converting speech file formats
  - GSE, 3-35
- Converting speech files for editing, 3-25
- Copying wordlists with the FlexWord toolkit, 4-41
- Corrupted speech disk
  - recovering, 1-3
- Creating filesystems
  - reference for, 1-3
- Creating FlexWord wordlists, 4-34
- Creating speech
  - overview of, 2-2
- Creating wordlists
  - FlexWord Speech Recognition, 4-30
- Custom DIPs for WholeWord Speech Recognition, 4-8
- Custom phrase tags
  - definition of, 1-5
- Custom recognition types for WholeWord Speech Recognition, 4-7
- Custom speech
  - producing with Script Builder, 2-10
- Custom Speech package
  - advantages of, 2-11
- Custom speech package, 2-11
  - advantages, 2-11

Custom vocabularies for WholeWord Speech Recognition, 4-7  
Customizing speech with Text-to-Speech, 2-12

---

## D

Deactivating wordlists, 4-43  
Decreasing the size of the speech filesystem  
  reference for, 1-3  
Default speech directory, 1-2  
Deleting words and wordlists from a FlexWord vocabulary, 4-40  
Designing a FlexWord Speech Recognition application, 4-26  
Developing speech  
  overview of, 2-2  
Digitizing speech  
  hiring AT&T, 2-19  
Digitizing speech phrases, 2-19  
Digitizing speech with Script Builder, 2-19  
Digitizing with the Graphical Speech Editor, 2-19  
Directory  
  default speech, 1-2  
  default speech filesystem, 1-6  
Disk  
  adding a second disk for speech, 1-3  
  recovering from a corrupted speech disk, 1-3  
Displaying file system attributes  
  reference for, 1-4  
doscp, 3-37  
Dual tone multifrequency support for WholeWord Speech Recognition, 4-11

---

## E

Editing a speech file with the Graphical Speech Editor, 3-15  
Editing speech files  
  using the Audio Works Station, 3-39  
Editing with Script Builder, 3-32  
Equipment  
  recording, 2-15  
erase command, 3-38  
Error messages  
  FlexWord toolkit, 4-36  
Escape sequences  
  sample Text-to-Speech application, D-8  
  Text-to-Speech, D-2

Examples of FlexWord applications, 5-5  
Experienced calling population, 4-45  
Extended vocabularies for FlexWord Speech Recognition, 4-24

---

## F

Falling inflection, 2-8  
Filesystem  
  changing the size of, 1-3  
  speech, 1-2  
Filesystem attributes  
  displaying, 1-4  
Filesystems  
  creating and removing, 1-3  
Fixed-Length connected digit  
  WholeWord Speech Recognition, 4-12  
FlexWord Editor, 4-34  
FlexWord Speech Recognition  
  additional requirements, 1-11  
  custom vocabulary service, 4-24  
  extended Vocabularies, 4-24  
  FlexWord toolkit, 4-24  
  introduction, 4-22  
  keyword spotting, 4-25  
  overview of, 1-11  
  phrase screening, 4-25  
  preparing an application, 4-26  
  using with Text-to-Speech, 5-7  
FlexWord Speech Recognition  
  activating wordlists, 4-43  
  building a FlexWord vocabulary, 5-3  
  caller error, 5-4  
  choosing a FlexWord vocabulary, 5-3  
  considering synonyms, 5-4  
  considering the sound of a word, 5-4  
  deactivating wordlists, 4-43  
  getting the most out of FlexWord, 5-3  
  installing wordlists, 4-42  
  prompting for input, 5-4  
  removing wordlists, 4-44  
  using with WholeWord Speech Recognition, 5-10  
  word length, 5-4  
  word meaning, 5-4  
FlexWord Speech Recognition accuracy, 4-45  
FlexWord Speech Recognition applications requiring  
  number-intensive wordlists, 5-5  
FlexWord Speech Recognition capacities, 4-23  
FlexWord Speech Recognition limitations, 4-25  
FlexWord Speech Recognition versus WholeWord Speech Recognition, 5-12  
FlexWord Toolkit  
  overview of, 1-11  
FlexWord toolkit  
  adding words and phrases to a vocabulary, 4-39  
  computing channel numbers, C-2

- configuring hardware, 4-31
- configuring the Text-to-Speech circuit card, 4-32
- connecting the phone line, 4-31
- copying wordlists, 4-41
- creating and modifying wordlists, 4-34
- deleting words and wordlists from a vocabulary, 4-40
- error messages, 4-36
- hardware requirements, 4-30
- software requirements, 4-30
- FlexWord toolkit for FlexWord Speech Recognition, 4-24
- FlexWord vocabulary administration, 4-42
- Framing
  - considering speech sounds, 2-8
  - examples of, 2-6
  - how to use in voice scripts, 2-6
  - placing words, 2-7
  - voiceless stops and sounds, 2-8

---

## G

- Getting the most out of Text-to-Speech, 5-6
- Glossary, GL-1
- Graphical Speech Editor, 2-13
  - accessing, 3-3
  - adjusting the volume, 3-22
  - channel numbers, 3-4
  - computing channel numbers, C-2
  - copying and pasting speech, 3-21
  - cutting and pasting speech, 3-19
  - exiting, 3-24
  - introduction, 3-2
  - listening to speech, 3-19
  - opening a file, 3-16
  - overview of producing and editing speech, 1-9
  - requirements of, 1-9
  - saving changes, 3-23
  - selecting a segment of speech, 3-18
  - setting the Marker, 3-17
  - speech energy, 3-12
  - speech envelope, 3-12
  - supported speech encoding formats, 3-2
  - undoing changes, 3-23
  - using the Mouse, 3-15

- Graphical Speech Editor edit menu, 3-8
- Graphical Speech Editor file menu, 3-6
- Graphical Speech Editor information bar, 3-13
- Graphical Speech Editor menu bar, 3-6
- Graphical Speech Editor options, 2-13
- Graphical Speech Editor scroll bar, 3-13
- Graphical Speech Editor speech display, 3-12
- Graphical Speech Editor title bar, 3-5
- Graphical Speech Editor tool bar, 3-9
- Graphical Speech Editor user interface, 3-4, 3-5
- Graphical Speech Editor voice menu, 3-7
- Graphical Speech Editor Window, 3-4
- gse command, 3-4
- GSE Features
  - Copy, 2-13
  - Cut, 2-13
  - Output volume, 2-13
  - Record, 2-13
- gse\_add command, 3-25, 3-37
- gse\_addpl, 3-25
- gse\_addpl command, 3-38
- gse\_copy command, 3-25
- gse\_copypl command, 3-25, 3-38

---

## H

- hardware installation of AYC9 SP
  - Text-to-Speech, 1-8
- Hardware requirements
  - FlexWord toolkit, 4-30
- Hiring a professional speaker
  - guidelines for, 2-9

---

## I

- Importing speech, 2-14
- Incoming text volume
  - relating to outgoing text volume, D-16
- Increasing the size of the speech filesystem
  - reference for, 1-3
- Inflection
  - falling, 2-8
  - medial, 2-8
  - rising, 2-8
- Inflections
  - recording, 2-17
  - speech, 2-8
  - writing the voice script, 2-8

Initiating the Graphical Speech Editor, 3-4  
Installing a FlexWord Speech Recognition wordlist, 4-42  
Installing and administering the mouse, 3-3, 4-33  
Installing speech in a C Application, 2-20  
Installing speech in a Script Builder application, 2-20  
Installing speech in a TSM application, 2-20  
Installing speech in an IRAPI application, 2-20  
International support for WholeWord Speech Recognition, 4-2  
Invoking the FlexWord Editor, 4-35

---

## K

Keyword spotting for FlexWord Speech Recognition, 4-25

---

## L

Limitations of FlexWord Speech Recognition, 4-25  
Listening to speech with the Graphical Speech Editor, 3-19  
Listening to the phonetic pronunciation of a word, 4-37

---

## M

Medial inflection, 2-8  
Menu-based FlexWord Speech Recognition applications, 5-5  
Models  
    definition of, 4-6  
Modifying FlexWord wordlists, 4-34  
Mouse  
    installing and administering, 3-3, 4-33  
    using with GSE, 3-2  
Mu-law PCM format, B-2

---

## N

Negative influences on Wholeword Speech Recognition accuracy, 4-16  
Numbers  
    recording, 2-16

---

## O

O.S. index  
    computing channel numbers, C-2

---

Opening a file with the FlexWord toolkit, 4-37  
Organization of book, xii  
Outgoing text volume  
    relating to incoming text volume, D-16  
Output Volume, 3-11

---

## P

Phonemes  
    definition of, 4-22  
phonetic spelling  
    viewing, 4-35  
Phrase numbers  
    definition of, 1-6  
Phrase rejection for WholeWord Speech Recognition, 4-9  
Phrase screening for FlexWord Speech Recognition, 4-25  
Phrase screening for WholeWord Speech Recognition, 4-9  
Phrase tag  
    definition of, 1-5  
Phrases  
    definition of, 1-5  
    digitizing, 2-19  
    long, 2-4  
    planning the voice script, 2-4  
    short, 2-4  
Planning a recording session  
    environmental conditions, 2-15  
    equipment specifications, 2-15  
Planning the voice script  
    suggestions for, 2-4  
    types of phrases, 2-4  
Play, 3-10  
Positive influences on Wholeword Speech Recognition accuracy, 4-13  
Predefined phrase tags  
    Script Builder, 1-6  
Preparing a FlexWord application, 4-26  
Prerecorded speech  
    using with Text-to-Speech, 5-8  
Processing recorded speech  
    overview of, 2-2  
Producing self-recorded custom speech, 2-10  
Professional speaker  
    hiring, 2-9  
Prompting for input  
    FlexWord Speech Recognition, 5-4  
Prompts  
    informative, 4-46  
    structure and design, 4-45  
Pulse Code Modulation, 3-2  
Pulse code modulation, B-2  
Purchasing a custom speech package, 2-11  
Purchasing an AT&T custom speech package  
    creating wordlists, 4-30

Purchasing the standard speech package, 2-11  
Purpose of book, xi

---

## R

Recognition confirmation for WholeWord Speech  
  Recognition, 4-9  
Recognizer, 4-12  
Record, 3-10  
Recording  
  alpha and numeric characters with inflections, 2-17  
  magnetic tape, 2-16  
Recording alpha characters, 2-16  
Recording equipment, 2-15  
Recording numbers, 2-16  
Recording session  
  environmental conditions, 2-15  
Recording speech  
  options for, 2-15  
Recording speech with a professional speaker, 2-15  
Recording speech with Script Builder, 2-18  
Recording spoken phrases on magnetic tape, 2-16  
Recovering from a corrupted speech disk  
  reference for, 1-3  
Rejection  
  FlexWord Speech Recognition, 4-25  
Relating the incoming text volume to outgoing text volume,  
  D-16  
Removing filesystems, 1-3  
Removing FlexWord Speech Recognition wordlists, 4-44  
Resources  
  related, xvi  
Retrieving files from speech pools, 3-26  
Retrieving speech files from the speech filesystem for  
  editing with GSE, 3-35  
Rising inflection, 2-8

---

## S

Sample application using escape sequences  
  Text-to-Speech, D-8  
Saving a file with the FlexWord toolkit, 4-38  
Screen displays in book, xv  
Script  
  definition of, xi  
Script Builder  
  editing speech files, 3-32  
  overview of producing speech, 1-7  
  recording speech, 2-18  
Script Builder application  
  installing speech, 2-20  
Second speech disk  
  adding, 1-3

Service marks, xvii  
Sharing speech, 2-13  
Silence delays  
  using escape sequences in Text-to-Speech, D-2  
Software requirements  
  FlexWord toolkit, 4-30  
Space requirements for speech phrases, 1-2  
Speaker  
  hiring a professional, 2-9  
Speaker independence for WholeWord Speech  
  Recognition, 4-9  
Speech  
  importing, 2-14  
  installing, 2-20  
  installing in a C application, 2-20  
  installing in an IRAPI application, 2-20  
  sharing, 2-13  
Speech administration tools, 1-3  
Speech development  
  hiring a professional speaker, 2-9  
  importing speech, 2-14  
  producing self-recorded custom speech, 2-10  
  purchasing the custom speech package, 2-11  
  selecting a method, 2-9  
  sharing speech, 2-13  
  using Text-to-Speech, 2-12  
  using the Graphical Speech Editor, 2-13  
Speech development features, 1-7  
Speech directory  
  default, 1-2  
  organization of, 1-2  
  specifying, 1-2  
Speech disk  
  recovering from a corrupted, 1-3  
Speech encoding formats, B-2  
Speech file formats, B-2  
Speech Files  
  definition of, 1-6  
Speech Filesystem  
  definition of, 1-6  
Speech filesystem, 1-2  
  changing the size of, 1-3  
Speech inflections, 2-8  
Speech phrase  
  coding rates, 1-2  
  space requirements, 1-2  
Speech phrases  
  digitizing, 2-19  
Speech pools  
  retrieving files from, 3-26  
Speech processing  
  overview of, 2-2  
Speech Recognition  
  limitations, 4-25  
Speech recognition  
  Bilingual speech recognition capabilities, 4-2  
  overview of, 1-10  
  sub-word, 4-22

Speech terminology, 1-5  
SPEECHDIR variable, 1-3  
Standard phrase tags  
    examples of, 1-5  
Standard speech package, 2-11  
Standard speech phrases, A-2  
Standard Vocabulary for WholeWord Speech Recognition,  
    4-2  
Stop, 3-10  
Storing speech files, 1-2  
Sub Band Coding, B-2  
Synonyms  
    FlexWord Speech Recognition, 5-4  
Synthesized speech, 1-8

---

## T

Talkfiles  
    definition of, 1-6  
    extension of, 1-6  
Telephone, 3-9  
Terminal keys  
    conventions used in book, xiv  
Text Classes  
    using escape sequences in Text-to-Speech, D-5  
Text-to-Speech  
    customizing, 2-12  
    examples of use, 1-8  
    getting the most out of Text-to-Speech, 5-6  
    how to use, 1-8  
    overview of producing speech, 1-8  
    requirements of, 1-8  
    software installation of, 1-8  
    using with FlexWord Speech Recognition, 5-7  
    using with prerecorded speech, 5-8  
Text-to-Speech application  
    changing the volume level, D-14  
Text-to-Speech functionality, 2-12  
Text-to-Speech technology, 2-12  
Third-party speech-editing system  
    Audio Works Station, 3-33  
Tone-paced digits for WholeWord Speech Recognition,  
    4-11  
Tool Bar  
    Output Volume, 3-11  
    Play, 3-10  
    Stop, 3-10  
    Telephone, 3-9  
Tool bar  
    record, 3-10  
Tools for speech administration, 1-3  
Touch tone  
    combining with WholeWord Speech Recognition, 5-2  
Trademarks, xvii  
Transaction  
    definition of, xi, 2-3

    determining, 2-3  
Transferring speech files  
    GSE, 3-37  
TSM application  
    installing speech, 2-20

---

## U

User input  
    conventions used in book, xiv  
Using escape sequences with Text-to-Speech, D-2  
Using FlexWord Speech Recognition and Text-to-Speech,  
    5-7  
Using Script Builder for editing, 3-32  
Using the FlexWord Toolkit to create wordlists, 4-30  
Using TTS and prerecorded speech together, 5-8  
Using WholeWord and FlexWord together, 5-10  
Using wordlists in FlexWord Speech Recognition scripts,  
    4-26

---

## V

Variable-Length connected digit  
    WholeWord Speech Recognition, 4-12  
Viewing the phonetic spelling of a word, 4-35  
Voice script  
    planning, 2-4  
    writing, 2-6  
Voiceless stops  
    writing the voice script, 2-8

---

## W

WholeWord custom vocabulary  
    overview of, 1-10  
WholeWord Speech Recognition  
    additional requirements, 1-10  
    associated DIP, 4-20  
    barge-in, 4-9  
    bilingual speech recognition capabilities, 4-2  
    computational processing, 4-20  
    connected digit recognition, 4-11  
    custom DIPs, 4-8  
    custom recognition types, 4-7  
    custom vocabularies, 4-7  
    dual tone multifrequency support, 4-11  
    fixed-length and variable-length connected digits, 4-12  
    how it works, 4-19  
    international support, 4-2  
    introduction, 4-2  
    overview of, 1-10  
    phrase screening, 4-9

- recognition confirmation, 4-9
- speaker independence, 4-9
- speech input from caller, 4-19
- standard vocabulary, 4-2
- the recognizer, 4-19
- word spotting, 4-8
- WholeWord Speech Recognition
  - combining with touch tone, 5-2
  - getting the most out of WholeWord, 5-2
  - using with FlexWord Speech Recognition, 5-10
- WholeWord Speech Recognition accuracy, 4-13
- WholeWord Speech Recognition grammars, 4-6
- WholeWord Speech Recognition models, 4-6
- WholeWord Speech Recognition types, 4-4
- WholeWord Speech Recognition versus FlexWord Speech Recognition, 5-12
- WholeWord standard vocabulary
  - overview of, 1-10
- wl\_copy command, 4-34
- wl\_edit command, 4-34
- wl\_init command, 4-34
- Word length
  - FlexWord Speech Recognition, 5-4
- Word meaning
  - FlexWord Speech Recognition, 5-4
- Word spotting
  - WholeWord Speech Recognition, 4-8
- Word spotting for WholeWord Speech Recognition, 4-8
- Wordlists
  - consolidation of, 4-27
  - creating, 4-30
  - effective construction of, 4-45
  - ineffective, 4-49
  - name tag characters, 4-23
  - using in scripts, 4-26
  - words per wordlist, 4-23
- Writing the script
  - framing, 2-6
- Writing the voice script
  - suggestions for, 2-6

